

ระบบทำรายงานงบกำไรขาดทุนอัตโนมัติ  
PROFIT AND LOSS REPORTS AUTOMATION



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา

วิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ปีการศึกษา 2563

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

# PROFIT AND LOSS REPORTS AUTOMATION



THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT  
OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR  
OF ENGINEERING IN INFORMATION ENGINEERING  
FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ACADEMIC YEAR 2020  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

หัวข้อโครงการ ระบบทำรายงานงบกำไรขาดทุนอัตโนมัติ  
รายชื่อนักศึกษา นางสาวหฤทัย อำนวยจจรูญ รหัสนักศึกษา 60011127  
ปริญญา วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา วิศวกรรมสารสนเทศ  
พ.ศ. 2563  
อาจารย์ที่ปรึกษาปริญญาานิพนธ์ ผศ.ดร.วันวิสา ชัชวงษ์

ปริญญาานิพนธ์ฉบับนี้ ได้รับการอนุมัติให้เป็นส่วนหนึ่งของการศึกษา ตามหลักสูตรวิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

หัวข้อโครงการ ระบบทำรายงานงบกำไรขาดทุนอัตโนมัติ  
รายชื่อนักศึกษา นางสาวหฤทัย อำนางจงจรูญ รหัสนักศึกษา 60011127  
ปริญญา วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา วิศวกรรมสารสนเทศ  
พ.ศ. 2563  
อาจารย์ที่ปรึกษาปริญญาานิพนธ์ ผศ.ดร.วันวิสา ชัชวงษ์

### บทคัดย่อ

รายงานงบกำไรขาดทุนเป็นส่วนสำคัญในการตรวจสอบสถานะของบริษัท ดูผลประกอบการ ตลอดจนการวางแผนบริหาร ช่วยให้ธุรกิจสามารถดำเนินการต่อไปได้ แต่ในปัจจุบันขั้นตอนการดำเนินการเพื่อสร้างรายงานงบกำไรขาดทุนของบริษัท ซีพี ออลล์ จำกัด (มหาชน) หรือ โลตัส กำลังประสบปัญหาในเรื่องของเวลาในกระบวนการจัดทำรายงานงบกำไรขาดทุน ต้องจัดทำรายงานทุกเดือน ใช้เวลาเดือนละ 48 ชั่วโมง ข้อมูลที่นำมาใช้นั้นมาจากหลายแหล่ง ต้องมีการคำนวณเพิ่มเติม ทำให้เกิดข้อผิดพลาดบ่อยครั้ง อีกทั้งข้อมูลที่นำมาใช้เป็นข้อมูลขนาดใหญ่ไม่สามารถจัดทำรายงานให้อยู่ภายใน Microsoft Excel ไฟล์เดียวได้ ปัญหาเหล่านี้ส่งผลให้เกิดการปรับปรุงและเปลี่ยนแปลงวิธีการสร้างรายงานแบบใหม่ที่สามารถประหยัดแรงงานคนและเวลาในการจัดทำรายงาน โดยปรับปรุงในการจัดทำรายงานเป็นแบบอัตโนมัติ จัดทำรายงานบนโปรแกรม Microsoft Power BI ซึ่งส่งผลให้เวลาที่ใช้ในการจัดทำรายงานลดลงเหลือเพียง 2 ชั่วโมงต่อเดือนเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



Project Title	Profit and Loss Report Automation	
Student	Miss Haruethai Amnatchongcharoon	Student ID. 60011127
Degree	Bachelor of Engineering	
Program	Information Engineering	
Year	2020	
Thesis Advisor	Asst.Prof.Dr. Vanvisa Chutchavong	

## ABSTRACT

The profit and loss reports are an important part of verifying the status of a company. See earnings as well as administrative planning, helping businesses to keep going. But nowadays, the process of generating profit and loss reports in which C.P. ALL PUBLIC COMPANY LIMITED (Lotus) is facing a problem with time in the reporting process. Financial department have to prepare a report every month. It takes 48 hours a month. The information used comes from many sources and have additional calculations are required, causing frequent errors. In addition, the data used as large data cannot be reported in a single Microsoft Excel file. These problems have resulted in improvements and changes to the new reporting methods that save manpower and time in the preparation of the report. By improving the reporting to be automatic. Generate reports in Microsoft Power BI, which results in a reduction in reporting time to 2 hours per month only.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## กิตติกรรมประกาศ

ข้าพเจ้าได้มีโอกาสรับผิดชอบและปฏิบัติหน้าที่ ที่บริษัทร้านค้าปลีกรายใหญ่ของประเทศอย่างแบรนด์ โลตัส ทำงานร่วมกับพนักงานในทีม IT - Data and Reporting ทำโปรเจกต์สร้างระบบจัดทำรายงานงบกำไรขาดทุนอัตโนมัติ ซึ่งข้าพเจ้าได้รับความรู้ และประสบการณ์ในการทำงานที่เป็นประโยชน์อย่างมากในอนาคต ปฏิญาณพันธะฉบับนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลือจากหลายฝ่าย ไม่ว่าจะเป็นพี่เลี้ยงในทีมโลตัส ที่คอยให้ความรู้ คำแนะนำ สอนทักษะต่าง ๆ ในการทำงาน อันได้แก่

- ขอขอบคุณ คุณอดิพร เจริญศรี ผู้อำนวยการเทคโนโลยีสารสนเทศ ที่จัดตั้งโครงการที่เปิดโอกาสให้นักศึกษาเข้ามามีส่วนร่วมในโปรเจกต์ของบริษัท
- ขอขอบคุณ คุณกัปด ปัญญาเวชมาณี หัวหน้าทีม IT - Data and Reporting ที่แนะนำระบบทั้งหมดในองค์กรให้อย่างละเอียด และคอยให้คำปรึกษามาโดยตลอด
- ขอขอบคุณ คุณนภัสพร ลิขิตส์จจากุล ที่เป็นที่ปรึกษา ให้คำแนะนำ รวมถึงให้ข้อมูลอันเป็นประโยชน์ ซึ่งเป็นส่วนสำคัญอย่างยิ่งที่ทำให้เกิดชิ้นงานนี้ขึ้นมา
- ขอขอบคุณ คุณนิรันดร์ รุ่งทอง ที่คอยให้ข้อมูล ให้คำแนะนำ และความรู้ทางด้านการเงิน อันเป็นประโยชน์อย่างมากในชิ้นงาน

ทั้งนี้ยังรวมถึงพนักงานอีกมากมายที่ดูแล เอาใจใส่ข้าพเจ้า ที่ทำให้การปฏิบัติงานที่โลตัสเต็มไปด้วยความสุข เป็นประสบการณ์ที่ข้าพเจ้าจะไม่ลืมเลือน

ข้าพเจ้าขอขอบคุณอาจารย์ที่ปรึกษา ผศ.ดร.วันวิสา ชัชวงษ์ ที่คอยให้คำแนะนำ คำปรึกษา คอยรับฟัง และช่วยเหลือปัญหาต่าง ๆ ในการทำโครงการครั้งนี้ สุดท้ายขอขอบคุณครอบครัวที่คอยสนับสนุนในสิ่งที่ข้าพเจ้าต้องการศึกษา เป็นที่ปรึกษา และให้กำลังใจข้าพเจ้ามาโดยตลอด

हत्यय अंनाजजजुणु

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

# สารบัญ

หน้า

บทคัดย่อภาษาไทย .....	I
บทคัดย่อภาษาอังกฤษ .....	II
กิตติกรรมประกาศ .....	III
สารบัญ .....	IV
สารบัญตาราง .....	VI
สารบัญรูป .....	VII
บทที่ 1 บทนำ .....	1
1.1 ความเป็นมาและความสำคัญของปัญหา .....	1
1.2 วัตถุประสงค์ของโครงการ .....	2
1.3 ขอบเขตของโครงการ .....	2
1.4 ขั้นตอนการดำเนินโครงการ .....	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ .....	3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง และเครื่องมือในการพัฒนาระบบ .....	4
2.1 รายงานงบกำไรขาดทุน .....	4
2.2 ฐานข้อมูล .....	5
2.3 เครือข่ายคอมพิวเตอร์ .....	5
2.4 เซิร์ฟเวอร์ .....	6
2.5 Batch File .....	7
2.6 เครื่องมือที่ใช้ .....	7
2.6.1 คอมพิวเตอร์โน้ตบุ๊ก .....	7
2.6.2 Window10 .....	7
2.6.3 Microsoft Excel .....	8
2.6.4 Jupyter Notebook .....	8
2.6.5 Microsoft SQL Server Management Studio 2017 .....	9
2.6.6 Microsoft Power BI Desktop .....	10
2.6.7 Microsoft Power BI Report Server .....	11
2.6.8 Task Scheduler .....	11
2.6.9 Dax Studio .....	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## สารบัญ (ต่อ)

	หน้า
บทที่ 3 วิธีการดำเนินโครงการ.....	13
3.1 ศึกษารายงานงบกำไรขาดทุนที่ใช้ในปัจจุบัน .....	13
3.1.1 กระบวนการจัดทำรายงานปัจจุบัน .....	13
3.1.2 ประเภทของรายงานงบกำไรขาดทุน .....	15
3.1.3 ข้อมูลที่นำมาใช้งาน.....	15
3.1.4 รายการประเภทของรายได้และค่าใช้จ่าย.....	17
3.2 เก็บรวบรวมสิ่งที่ผู้ใช้งานต้องการ และปัญหาที่ผู้ใช้งานประสบ.....	18
3.3 คิด วิเคราะห์ หาทางแก้ไขปัญหา .....	18
3.4 ออกแบบและพัฒนาระบบ .....	19
3.4.1 ออกแบบและสร้างฐานข้อมูล .....	19
3.4.2 ออกแบบและเขียนโปรแกรมสร้างการไหลของข้อมูล.....	29
3.4.3 ออกแบบและพัฒนาส่วนอัตโนมัติ.....	54
3.5 สร้างหน้ารายงาน.....	57
3.5.1 โหลดข้อมูลเข้า Microsoft Power BI.....	57
3.5.2 สร้างเส้นความสัมพันธ์ระหว่างตาราง .....	65
3.5.3 จัดทำรายงานงบกำไรขาดทุน .....	65
บทที่ 4 ผลการดำเนินโครงการ.....	67
4.1 วิธีการเข้าถึงรายงานงบกำไรขาดทุน .....	67
4.2 รายงานงบกำไรขาดทุน .....	68
4.2.1 หน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับแสดงผลบน server.....	68
4.2.2 หน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับการ export ข้อมูล .....	69
4.2.3 หน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับป้อนลงกระดาษ.....	70
บทที่ 5 สรุปผลการวิจัยแล้วขอเสนอแนะ .....	71
5.1 สรุปผล .....	71
เอกสารอ้างอิง.....	73
ภาคผนวก.....	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

# สารบัญตาราง

หน้า

ตารางที่ 5.1 ตารางเปรียบเทียบประสิทธิภาพระหว่างกระบวนการสร้างหน้ารายงานแบบเก่าและแบบใหม่ .....71



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



# สารบัญรูป

หน้า

ภาพที่ 2.1 ภาพแสดงตัวอย่างรายงานงบกำไรขาดทุนอย่างง่าย .....	4
ภาพที่ 2.2 ภาพแสดงเครือข่ายคอมพิวเตอร์ .....	5
ภาพที่ 2.3 ภาพแสดงการเชื่อมต่อระหว่างผู้ใช้งานกับเซิร์ฟเวอร์ .....	6
ภาพที่ 2.4 ภาพแสดงสัญลักษณ์ของโปรแกรม Microsoft Excel .....	8
ภาพที่ 2.5 ภาพแสดงหน้าต่างของ Jupyter Notebook .....	8
ภาพที่ 2.6 ภาพแสดงหน้าต่างของโปรแกรม Microsoft SQL server management studio .....	9
ภาพที่ 2.7 ภาพแสดงหน้าต่างของ Microsoft Power BI .....	10
ภาพที่ 2.8 ภาพแสดงหน้าต่างของ Microsoft Power BI Server .....	11
ภาพที่ 2.9 ภาพแสดงหน้าต่างของ Task Scheduler .....	11
ภาพที่ 2.10 ภาพแสดงหน้าต่างของ Dax Studio .....	12
ภาพที่ 3.1 ภาพแสดงกระบวนการจัดทำรายงานปัจจุบัน .....	13
ภาพที่ 3.2 ภาพแสดงหน้ารายงานงบกำไรขาดทุนที่ใช้ในปัจจุบัน .....	14
ภาพที่ 3.3 ภาพแสดงข้อมูล Trial balance .....	15
ภาพที่ 3.4 ภาพแสดงข้อมูล Budget .....	16
ภาพที่ 3.5 ภาพแสดงข้อมูล DC .....	16
ภาพที่ 3.6 ภาพแสดงข้อมูล HO .....	16
ภาพที่ 3.7 ภาพแสดงโครงสร้างตาราง Actual data .....	20
ภาพที่ 3.8 ภาพแสดงข้อมูล Actual data ในโปรแกรม SSMS .....	20
ภาพที่ 3.9 ภาพแสดงโครงสร้างตาราง Target data .....	22
ภาพที่ 3.10 ภาพแสดงข้อมูล Target data ในโปรแกรม SSMS .....	22
ภาพที่ 3.11 ภาพแสดงโครงสร้างตาราง DC data .....	23
ภาพที่ 3.12 ภาพแสดงข้อมูล DC data ในโปรแกรม SSMS .....	24
ภาพที่ 3.13 ภาพแสดงโครงสร้างตาราง HO data .....	25
ภาพที่ 3.14 ภาพแสดงข้อมูล HO data ในโปรแกรม SSMS .....	25
ภาพที่ 3.15 ภาพแสดงโครงสร้างตาราง P&L snapshot .....	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



## สารบัญรูป (ต่อ)

หน้า

ภาพที่ 3.16 ภาพแสดงโครงสร้างตาราง P&L allocated snapshot .....	27
ภาพที่ 3.17 ภาพแสดงโครงสร้างตาราง P&L allocated snapshot .....	29
ภาพที่ 3.18 ภาพแสดงโครงสร้างตาราง P&L allocated snapshot .....	29
ภาพที่ 3.19 ภาพแสดงการไหลของข้อมูลในการจัดทำรายงานงบกำไรขาดทุน .....	30
ภาพที่ 3.20 ภาพแสดงวิธีการรับข้อมูลดิบ .....	30
ภาพที่ 3.21 ภาพแสดงการไหลของข้อมูลจากโพลเดอร์ส่วนกลางไปยังเซิร์ฟเวอร์ .....	31
ภาพที่ 3.22 ภาพแสดงโค้ด framework (1) .....	31
ภาพที่ 3.23 ภาพแสดงโค้ด framework (2) .....	32
ภาพที่ 3.24 ภาพแสดงโค้ด framework (3) .....	32
ภาพที่ 3.25 ภาพแสดงโค้ด framework (4) .....	32
ภาพที่ 3.26 ภาพแสดงโค้ด framework (5) .....	32
ภาพที่ 3.27 ภาพแสดงโค้ด framework (6) .....	33
ภาพที่ 3.28 ภาพแสดงโค้ด framework DC (1) .....	33
ภาพที่ 3.29 ภาพแสดงโค้ด framework DC (2) .....	34
ภาพที่ 3.30 ภาพแสดงโค้ด framework DC (3) .....	34
ภาพที่ 3.31 ภาพแสดงโค้ด framework DC (4) .....	34
ภาพที่ 3.32 ภาพแสดงโค้ด framework DC (5) .....	35
ภาพที่ 3.33 ภาพแสดงโค้ด framework DC (6) .....	35
ภาพที่ 3.34 ภาพแสดงโค้ด framework DC (7) .....	35
ภาพที่ 3.35 ภาพแสดงโค้ด framework HO (1) .....	36
ภาพที่ 3.36 ภาพแสดงโค้ด framework HO (2) .....	36
ภาพที่ 3.37 ภาพแสดงโค้ด framework HO (3) .....	36
ภาพที่ 3.38 ภาพแสดงโค้ด framework HO (4) .....	37
ภาพที่ 3.39 ภาพแสดงโค้ด framework HO (5) .....	37
ภาพที่ 3.40 ภาพแสดงโค้ด framework HO (6) .....	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational VIII only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## สารบัญรูป (ต่อ)

หน้า

ภาพที่ 3.41 ภาพแสดงโค้ด framework HO (7).....	37
ภาพที่ 3.42 ภาพแสดงการไหลของข้อมูลภายในเซิร์ฟเวอร์ฐานข้อมูล .....	38
ภาพที่ 3.43 ภาพแสดงโค้ดของฟังก์ชัน GetAccountMappingForPL (1).....	39
ภาพที่ 3.44 ภาพแสดงโค้ดของฟังก์ชัน GetAccountMappingForPL (2).....	39
ภาพที่ 3.45 ภาพแสดงโค้ดของฟังก์ชัน GetFormatMappingForPL (1).....	40
ภาพที่ 3.46 ภาพแสดงโค้ดของฟังก์ชัน GetFormatMappingForPL (2).....	40
ภาพที่ 3.47 ภาพแสดงโค้ดของ spTBByPeriod (1).....	41
ภาพที่ 3.48 ภาพแสดงโค้ดของ spTBByPeriod (2).....	41
ภาพที่ 3.49 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (1).....	42
ภาพที่ 3.50 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (2).....	42
ภาพที่ 3.51 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (3).....	43
ภาพที่ 3.52 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (4).....	44
ภาพที่ 3.53 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (5).....	45
ภาพที่ 3.54 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (6).....	46
ภาพที่ 3.55 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (7).....	47
ภาพที่ 3.56 ภาพแสดงโค้ดของ spLoadTBToPLSnapshot .....	48
ภาพที่ 3.57 ภาพแสดงโค้ดของ spAllocateTBByPeriod.....	49
ภาพที่ 3.58 ภาพแสดงโค้ดของ spAllocateTBByPeriodByFormat (1).....	50
ภาพที่ 3.59 ภาพแสดงโค้ดของ spAllocateTBByPeriodByFormat (2).....	50
ภาพที่ 3.60 ภาพแสดงโค้ดของ spAllocateTBByPeriodByFormat (3).....	51
ภาพที่ 3.61 ภาพแสดงโค้ดของ spAllocateTBByPeriodByFormat (4).....	51
ภาพที่ 3.62 ภาพแสดงโค้ดของ spAllocateTBByPeriodByFormat (5).....	52
ภาพที่ 3.63 ภาพแสดงโค้ดของ fpa_channel_pl_snapshot_v .....	53
ภาพที่ 3.64 ภาพแสดงโค้ดของ fpa_channel_pl_allocate_snapshot_v.....	53

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

# สารบัญรูป (ต่อ)

หน้า

ภาพที่ 3.66 ภาพแสดงการตั้งค่าในโปรแกรม Task Scheduler.....	55
ภาพที่ 3.67 ภาพแสดงโค้ด Python ในส่วนอัตโนมัติ (1).....	55
ภาพที่ 3.68 ภาพแสดงโค้ด Python ในส่วนอัตโนมัติ (2).....	56
ภาพที่ 3.69 ภาพแสดงโค้ด Python ในส่วนอัตโนมัติ (3).....	56
ภาพที่ 3.70 ภาพแสดงการตั้งค่าใน Microsoft Power BI Server.....	57
ภาพที่ 3.71 ภาพแสดงตารางที่นำเข้า Microsoft Power BI.....	57
ภาพที่ 3.72 ภาพแสดงการนำเข้าข้อมูลของตาราง CustomAccounts.....	58
ภาพที่ 3.73 ภาพแสดงการนำเข้าข้อมูลของตาราง DIM-Account.....	59
ภาพที่ 3.74 ภาพแสดงการนำเข้าข้อมูลของตาราง DIM-Store.....	60
ภาพที่ 3.75 ภาพแสดงการนำเข้าข้อมูลของตาราง TB แบบ KPI.....	61
ภาพที่ 3.76 ภาพแสดงการนำเข้าข้อมูลของตาราง TB แบบ Full set.....	62
ภาพที่ 3.77 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Template-Store-1-Order.....	62
ภาพที่ 3.78 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Template-Store-1-1-Order.....	63
ภาพที่ 3.79 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Template-Store-1-1-1-Order.....	63
ภาพที่ 3.80 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Store.....	64
ภาพที่ 3.81 ภาพแสดงการนำเข้าข้อมูลของตาราง M-Format-Hierarchy.....	64
ภาพที่ 3.82 ภาพแสดงความสัมพันธ์ของตารางใน Microsoft Power BI.....	65
ภาพที่ 3.83 ภาพแสดงตัวอย่างหน้ารายงานงบกำไรขาดทุน.....	65
ภาพที่ 3.84 ภาพแสดงโค้ดที่ใช้หาค่าเปอร์เซ็นต์.....	66
ภาพที่ 4.1 ภาพแสดงหน้าต่างของ Microsoft Power BI Server.....	67
ภาพที่ 4.2 ภาพแสดงระดับของข้อมูลสำหรับแสดงผลบน server.....	68
ภาพที่ 4.3 ภาพแสดงหน้ารายงานงบกำไรขาดทุน.....	68
ภาพที่ 4.4 ภาพแสดงระดับของข้อมูลสำหรับ export.....	69
ภาพที่ 4.5 ภาพแสดงหน้ารายงานงบกำไรขาดทุนสำหรับการ export.....	69
ภาพที่ 4.6 ภาพแสดงหน้ารายงานงบกำไรขาดทุนสำหรับปรี้น A3.....	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดเบี่ยงเบนเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

บริษัท เอก-ชัย ดีสทริบิวชั่น ซิสเทม จำกัด ก่อตั้งโลตัส ซูเปอร์เซ็นเตอร์ ขึ้นในวันที่ 28 ตุลาคม พุทธศักราช 2537 ดำเนินการธุรกิจค้าปลีก ภายใต้การบริหารงานของเครือเจริญโภคภัณฑ์ กระทั่งต่อมาในปี พุทธศักราช 2541 กลุ่มเทสโก้ (Tesco) กลุ่มค้าปลีกชั้นนำจากสหราชอาณาจักรเข้ามาบริหารงานแทน จึงได้เปลี่ยนชื่อจาก “โลตัส ซูเปอร์เซ็นเตอร์” เป็น “เทสโก้ โลตัส” บริหารงานอย่างราบรื่นตลอดมาจนถึงช่วงปลายปี พุทธศักราช 2563 บริษัท ซี.พี. รีเทล โฮลดิ้ง จำกัด ซึ่งถือหุ้นโดย บมจ.ซีพี ออลล์ (CPALL) สัดส่วน 40% บมจ.เจริญโภคภัณฑ์โฮลดิ้ง สัดส่วน 40% และบริษัท ซี.พี. เมอร์แกนไคซิง จำกัด ซึ่งเป็นบริษัทลูกของ CPF สัดส่วน 20% ได้เข้าซื้อกิจการ บริษัท เทสโก้ สโตร์ส (ประเทศไทย) จำกัด และ Tesco Store (Malaysia) เปลี่ยนชื่อและปรับเปลี่ยนแบรนด์เทสโก้จาก “เทสโก้ โลตัส” (TESCO Lotus) เป็น “โลตัส” (Lotus) โดยทางเครือซีพีได้ปรับรูปแบบแบรนด์โลตัสครั้งใหญ่ตามกฎหมายของการโอนถ่ายกิจการจากกลุ่มเทสโก้ ประเทศอังกฤษ และเพื่อให้สอดคล้องกับยุทธศาสตร์การบริหารงานของเครือซีพีที่ต้องการให้โลตัสมีความทันสมัยมากขึ้น หลังจากที่โลตัสเปิดให้บริการในไทยมานานกว่า 27 ปี โดยนำร่องปรับแบรนด์ครั้งใหญ่และทยอยรีโนเวทสาขาเลียบบางส่วนตามอินทราฯ สาขา อ่อนนุช และสาขาพระราม 4 ก่อนจะทยอยปรับสาขาอื่น ๆ ทั้งในประเทศไทยและมาเลเซียต่อไป

นับเป็นเวลากว่า 27 ปีที่เทสโก้โลตัสให้บริการลูกค้า 17 ล้านคนต่อสัปดาห์ ด้วยสาขาทั้งหมดทั่วประเทศ ประมาณ 2,000 สาขา รวมถึงแพลตฟอร์มทางออนไลน์ และยังคงมีการขยายตัวของกิจการเพิ่มขึ้นทุกปีอีกด้วย

เนื่องจากเทสโก้โลตัสเป็นบริษัทค้าปลีกขนาดใหญ่ มีรายได้และค่าใช้จ่ายที่มาจากหลากหลายส่วน จึงทำให้การบริหารจัดการดำเนินธุรกิจนี้เป็นเรื่องละเอียดอ่อนอย่างมาก ทางฝ่ายบริหารจะต้องทราบว่ารายได้เกิดจากส่วนใดมากที่สุด หรือรายจ่ายมาจากส่วนใด สามารถตัดรายจ่ายส่วนนั้นออกไปได้หรือไม่ เพื่อให้ได้กำไรเพิ่มมากขึ้น ทางฝ่ายการเงินจึงมีการจัดทำรายงานทางการเงินขึ้นมา ซึ่งหนึ่งในรายงานที่สำคัญคือ รายงานงบกำไรขาดทุน

รายงานงบกำไรขาดทุน เป็นรายงานทางการเงินของกิจการที่จัดทำขึ้นอย่างมีแบบแผน โดยมีวัตถุประสงค์เพื่อแสดง ฐานะทางการเงิน ผลการดำเนินงาน ตอบสนองต่อผู้ซึ่งบการเงิน ลักษณะรายงานงบกำไรขาดทุนที่มีคุณภาพต้องประกอบด้วย ความเข้าใจได้ ความเกี่ยวข้องกับการตัดสินใจ ความเชื่อถือได้ และการเปรียบเทียบกันได้ ข้อมูลทางการเงิน ที่แสดงในงบกำไรขาดทุน แสดงถึงผลกระทบทางการเงินจากเหตุการณ์ในอดีต การจัดทำงบการเงินและการนำเสนองบการเงิน ต้องชัดเจน ครบคลุม มีความถูกต้องตามที่ควรเป็นไปตามมาตรฐานการบัญชีที่รับรองทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



จากกระบวนการในการจัดทำรายงานงบกำไรขาดทุนที่ได้กล่าวไว้ข้างต้น ปัญหาและอุปสรรคที่เกิดขึ้นสามารถแบ่งออกได้ 3 อย่างด้วยกัน คือ

- ปัญหาเรื่องเวลา เนื่องจากข้อมูลที่น่ามาจากฐานข้อมูลไม่สามารถนำมาใช้ได้ทันที ต้องมีการแปลงข้อมูลด้วยวิธีการต่าง ๆ หลายขั้นตอน เพื่อให้ข้อมูลอยู่ในมุมมองหรือรูปแบบที่ต้องการ จึงทำให้ต้องใช้เวลาในการจัดทำถึง 48 ชั่วโมง ต่อเดือน
- ปัญหาเรื่องความถูกต้องแม่นยำ เนื่องจากรายงานงบกำไรขาดทุนถูกจัดทำโดยพนักงานฝ่ายการเงินในทุกขั้นตอน ในบางครั้งจึงเกิดข้อผิดพลาดที่เกิดจากมนุษย์ขึ้น ทำให้ข้อมูลผิดเพี้ยนไปจากความเป็นจริง
- ปัญหาเรื่องข้อจำกัดของโปรแกรม Microsoft Excel ที่ใช้ในการจัดทำและแสดงหน้ารายงานงบกำไรขาดทุนเนื่องจากข้อมูลที่น่ามาใช้ทำรายงานงบกำไรขาดทุนนั้นเป็นข้อมูลขนาดใหญ่ จึงต้องมีการแบ่งรายงานงบกำไรขาดทุนออกเป็น 4 ไฟล์ จึงจะสามารถใช้งานได้

จากสาเหตุเหล่านี้ทางบริษัทจึงได้มีการมอบหมายให้ปรับปรุงและพัฒนากระบวนการในการสร้างรายงานงบกำไรขาดทุน เพื่อความสะดวกและรวดเร็วในการนำไปใช้งาน

## 1.2 วัตถุประสงค์ของโครงการ

- เพื่อพัฒนาระบบรายงานงบกำไรขาดทุนให้เป็นแบบอัตโนมัติ
- เพื่อลดเวลาในการจัดทำรายงานงบกำไรขาดทุน
- เพื่อลดจำนวนทรัพยากรคนที่ต้องใช้ในการจัดทำรายงานงบกำไรขาดทุน
- เพื่อให้ข้อมูลในรายงานงบกำไรขาดทุนมีความถูกต้องแม่นยำมากยิ่งขึ้น
- เพื่อให้ผู้ใช้งานรายงานงบกำไรขาดทุนได้รับข้อมูลที่ต้องการ สามารถเข้าถึงรายงานงบกำไรขาดทุนได้อย่างสะดวก และใช้งานได้ง่าย

## 1.3 ขอบเขตของโครงการ

- เก็บข้อมูลจากผู้ใช้งานและฝ่ายอื่น ๆ ที่เกี่ยวข้อง นำข้อมูลไปวิเคราะห์ออกแบบระบบการสร้างรายงานงบกำไรขาดทุน
- ออกแบบระบบฐานข้อมูลที่รองรับกับข้อมูลการเงินเพื่อนำไปสร้างระบบรายงานงบกำไรขาดทุน
- ออกแบบรูปแบบการแสดงผลและจัดทำรายงานงบกำไรขาดทุน ให้ตรงตามความต้องการของผู้ใช้งาน
- กำหนดสิทธิ์ของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## 1.4 ขั้นตอนการดำเนินโครงการ

- ศึกษารายงานงบกำไรขาดทุนที่ใช้ในปัจจุบัน
- เก็บรวบรวมสิ่งที่ผู้ใช้งานต้องการ และปัญหาที่ผู้ใช้งานประสบ
- คิด วิเคราะห์ หาทางแก้ไขปัญหา
- ออกแบบหน้าตาของหน้ารายงานงบกำไรขาดทุน นำเสนอให้กับผู้ใช้งาน และปรับปรุงให้เป็นไปตามที่ผู้ใช้งานต้องการ
- สร้างระบบฐานข้อมูลเพื่อใช้ในการเก็บข้อมูลที่ต้องนำมาใช้สร้างรายงานงบกำไรขาดทุน
- พัฒนาระบบตามที่ได้ออกแบบไว้
- จัดทำรายงานงบกำไรขาดทุน
- ออกแบบ และจัดทำให้ระบบเป็นแบบอัตโนมัติ
- ทดสอบระบบ และแก้ไขข้อบกพร่อง
- ตรวจสอบความถูกต้องของข้อมูล
- กำหนดสิทธิให้กับผู้ใช้งาน
- เสนอแนะแนวทางเพื่อเพิ่มประสิทธิภาพ
- สรุปผล

## 1.5 ประโยชน์ที่คาดว่าจะได้รับ

- ลดจำนวนพนักงานในการจัดทำรายงานงบกำไรขาดทุน
- ใช้เวลาในการจัดทำรายงานงบกำไรขาดทุนลดลง
- ข้อมูลที่ได้ถูกต้อง ครบถ้วน สมบูรณ์
- ได้รับความรู้เกี่ยวกับการทำงานของฝ่ายการเงิน
- ได้รับความรู้เกี่ยวกับรายงานงบกำไรขาดทุน ทั้งวิธีการคำนวณ และความหมายของข้อมูล
- ได้รับความรู้และความเข้าใจในเรื่องของขั้นตอนในการทำงานจริง
- ได้รับความรู้ในการออกแบบระบบฐานข้อมูล
- ได้รับความรู้เกี่ยวกับการใช้โปรแกรมต่าง ๆ ในการจัดการข้อมูลขนาดใหญ่
- ได้รับความรู้เกี่ยวกับการจัดการเวลาให้มีประสิทธิภาพ เพื่อให้งานสำเร็จตามกำหนดการ
- ได้รับความรู้เกี่ยวกับการจัดการแก้ปัญหาเฉพาะหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



## บทที่ 2

# ทฤษฎีที่เกี่ยวข้อง และเครื่องมือในการพัฒนาระบบ

ในบทนี้จะอธิบายรายละเอียดเกี่ยวกับข้อมูลพื้นฐานที่ควรทราบในการทำโครงการ รวมถึงเทคโนโลยีและเครื่องมือต่าง ๆ ที่ใช้ในการพัฒนาระบบ

### 2.1 รายงานงบกำไรขาดทุน

รายงานงบกำไรขาดทุน (Profit and Loss Statement) คือ รายงานที่แสดงถึงผลการดำเนินงานของกิจการ หรือความสำเร็จในการดำเนินงานในช่วงระยะเวลาหนึ่ง เพื่อให้ทางผู้ประกอบการนำมาประเมินผลการดำเนินงานในอดีต การคาดคะเนผลการดำเนินงานในอนาคต และประเมินความเสี่ยงของธุรกิจ

ในรายงานจะไม่แสดงรายการรายได้หรือค่าใช้จ่าย ที่ไม่สามารถวัดมูลค่าได้อย่างน่าเชื่อถือ วิธีการบัญชีที่ใช้มีผลต่อยอดผลกำไร และการวัดผลกำไรต้องใช้ดุลยพินิจของตนเอง



ร้านอารียา  
งบกำไรขาดทุน  
สำหรับระยะเวลา 1 ปี สิ้นสุดวันที่ 31 ธันวาคม 25X6

รายได้		
รายได้จากการขาย (รายละเอียด 1)		354,800
ค่าใช้จ่าย		
ต้นทุนขาย (รายละเอียด 2)		157,080
กำไรขั้นต้น		197,720
บวก รายได้ค่าเช่า		5,250
กำไรก่อนหักค่าใช้จ่าย		202,970
หัก ค่าใช้จ่ายในการขายและการบริหาร (รายละเอียด 3)		99,200
กำไรก่อนหักต้นทุนการเงินและภาษีเงินได้		103,770
หัก ดอกเบี้ยจ่าย		1,725
กำไรก่อนหักภาษีเงินได้		102,045
หัก ภาษีเงินได้		XX
กำไรสุทธิ		102,045

ภาพที่ 2.1 ภาพแสดงตัวอย่างรายงานงบกำไรขาดทุนอย่างง่าย

ผลกำไร (ขาดทุน) คือ รายการที่เป็นไปตามค่านิยมของรายได้ (ค่าใช้จ่าย) และอาจเกิดจากกิจกรรมตามปกติของกิจการหรือไม่ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

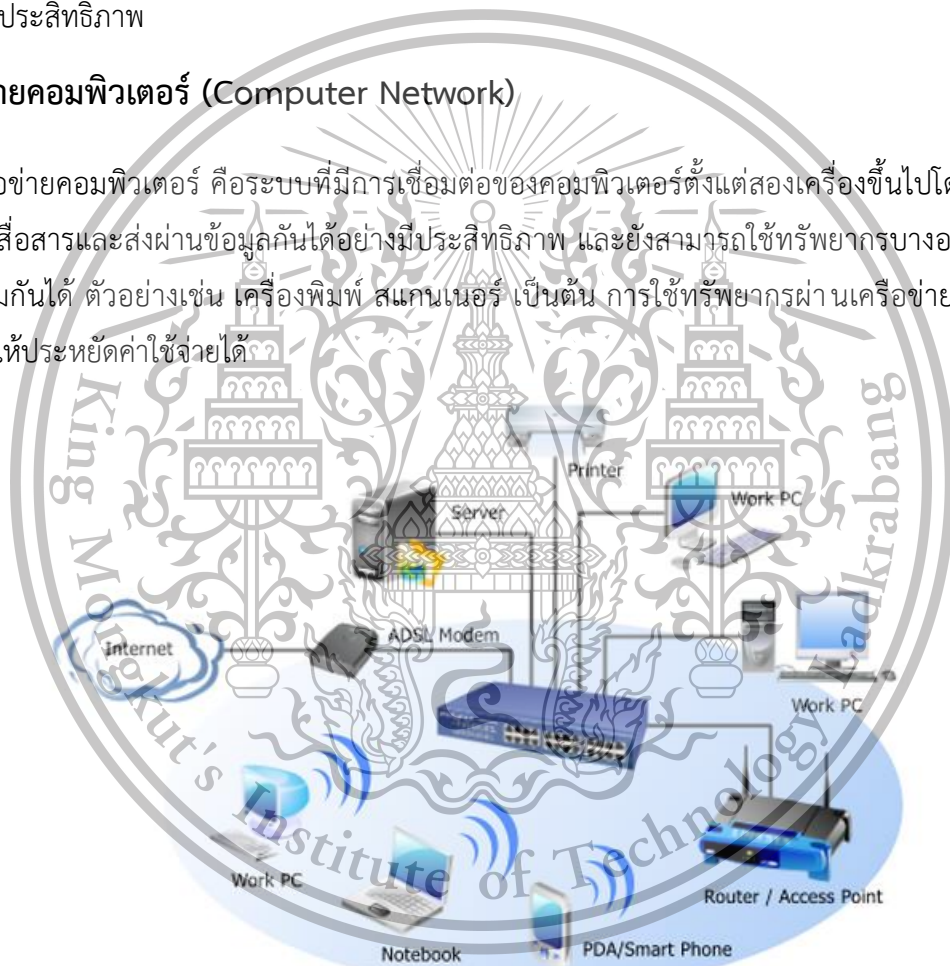
## 2.2 ฐานข้อมูล (Database)

ฐานข้อมูล คือ ข้อมูลต่าง ๆ ถูกเก็บรวบรวมไว้ โดยไม่จำเป็นว่าข้อมูลทั้งหมดต้องเก็บรวบรวมไว้ในที่เดียวกัน สามารถแยกเก็บหลาย ๆ ที่ได้

ระบบฐานข้อมูล (Database System) คือ ระบบที่รวมข้อมูลต่าง ๆ หลายเพิ่มข้อมูลที่มีความเกี่ยวข้องกันอย่างชัดเจน ผู้ใช้งานสามารถใช้งาน ดูแลรักษาและป้องกันข้อมูลได้อย่างมีประสิทธิภาพ โดยมีซอฟต์แวร์ที่เปรียบเสมือนเป็นสื่อกลางระหว่างผู้ใช้งานกับโปรแกรม เรียกว่า ระบบจัดการฐานข้อมูล ทำให้ผู้ใช้งานเข้าถึงข้อมูลได้ง่าย และมีประสิทธิภาพ

## 2.3 เครือข่ายคอมพิวเตอร์ (Computer Network)

เครือข่ายคอมพิวเตอร์ คือระบบที่มีการเชื่อมต่อของคอมพิวเตอร์ตั้งแต่สองเครื่องขึ้นไปโดยใช้สื่อกลาง และสามารถสื่อสารและส่งผ่านข้อมูลกันได้อย่างมีประสิทธิภาพ และยังสามารถใช้ทรัพยากรบางอย่างที่มีอยู่ในเครือข่ายร่วมกันได้ ตัวอย่างเช่น เครื่องพิมพ์ สแกนเนอร์ เป็นต้น การใช้ทรัพยากรผ่านเครือข่ายคอมพิวเตอร์สามารถช่วยให้ประหยัดค่าใช้จ่ายได้



ภาพที่ 2.2 ภาพแสดงเครือข่ายคอมพิวเตอร์

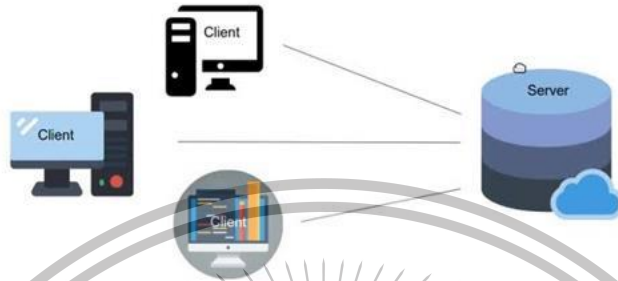
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## 2.4 เซิร์ฟเวอร์ (Server)

เซิร์ฟเวอร์ คือ เครื่องคอมพิวเตอร์หรือระบบปฏิบัติการหรือโปรแกรมคอมพิวเตอร์ที่ทำหน้าที่ให้บริการแก่เครื่องคอมพิวเตอร์หรือโปรแกรมคอมพิวเตอร์อื่นที่เป็นลูกข่ายในระบบเครือข่าย



ภาพที่ 2.3 ภาพแสดงการเชื่อมต่อระหว่างผู้ใช้งานกับเซิร์ฟเวอร์

เครื่องคอมพิวเตอร์ที่เป็นเซิร์ฟเวอร์จะแบ่งออกเป็น 2 แบบด้วยกัน

- 1) Rack server จะมีลักษณะเป็นแท่งสี่เหลี่ยมยาว ๆ
- 2) Tower server จะมีลักษณะเหมือนเครื่องคอมพิวเตอร์ทั่วไป (Personal Computer)  
ตัวอย่างโปรแกรมคอมพิวเตอร์ ที่ทำหน้าที่เป็นเซิร์ฟเวอร์ ตัวอย่างเช่น
  - Web Server คือโปรแกรมคอมพิวเตอร์ที่ทำหน้าที่ให้บริการเว็บ เช่น Apache
  - Mail Server คือโปรแกรมคอมพิวเตอร์ที่ทำหน้าที่ให้บริการอีเมล เช่น Postfix
  - DNS Server คือโปรแกรมคอมพิวเตอร์ที่ทำหน้าที่ให้บริการโดเมนเนม เช่น Bind9
  - Database Server คือโปรแกรมคอมพิวเตอร์ที่ทำหน้าที่ให้บริการฐานข้อมูล เช่น MySQL

สำหรับระบบปฏิบัติการที่นิยมใช้เป็นเซิร์ฟเวอร์ ได้แก่

- 1) Linux ที่ได้รับความนิยมได้แก่ Ubuntu
- 2) Window ที่ได้รับความนิยมได้แก่ Window server 2003
- 3) Unix เป็นระบบปฏิบัติการที่เก่าแก่ระบบหนึ่ง ที่ยังใช้ในปัจจุบันได้แก่ BSD

โดยปกติแล้วเครื่องคอมพิวเตอร์ที่ถูกใช้เป็นเซิร์ฟเวอร์หนึ่งเครื่อง จะสามารถติดตั้งระบบปฏิบัติการเพื่อใช้เป็นเซิร์ฟเวอร์ได้หนึ่งระบบปฏิบัติการ แต่สามารถลงโปรแกรมคอมพิวเตอร์ได้หลายชนิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ประเภทของเซิร์ฟเวอร์นั้นแบ่งออกเป็น 4 ประเภท โดยแบ่งตามลักษณะการทำงานเป็นหลัก ได้แก่

- 1) File Server มีหน้าที่ในการจัดเก็บไฟล์ ผู้ใช้งานสามารถฝากไฟล์ไว้ใน File Server ได้
- 2) Print Server มีหน้าที่ในการเชื่อมต่อเครื่องพิมพ์ให้สามารถใช้งานกับเครื่องคอมพิวเตอร์ที่เป็นลูกข่าย เพื่อประหยัดทรัพยากร
- 3) Database Server มีหน้าที่ในการรันระบบที่เป็นฐานข้อมูล DBMS (Database Management System) เช่น SQL ซึ่งเป็นโปรแกรมฐานข้อมูลและตัวจัดการฐานข้อมูล
- 4) Application Server มีหน้าที่ในการรันโปรแกรมประยุกต์ โดยทำงานสอดคล้องกับผู้ใช้ งาน ตัวอย่างเช่น Web Server (รัน Web Server Program เช่น Apache)

## 2.5 Batch File

เป็นไฟล์ที่รวมชุดคำสั่งภายในและภายนอกที่สั่งให้คอมพิวเตอร์ทำงานตามที่ต้องการ โดยไฟล์ที่ถูกสร้างขึ้นจะมีนามสกุลของไฟล์เป็น .bat โดยแบทช์ไฟล์นี้สามารถติดต่อกับแบทช์ไฟล์อื่นหรือรับ-ส่งตัวแปร เข้าหรือออกจากแบทช์ไฟล์ได้เช่นเดียวกับการเขียนโปรแกรมทุกประการ แต่โปรแกรมที่เป็นแบทช์ไฟล์นี้ ระบบคอมพิวเตอร์สามารถประมวลผลคำสั่งได้ทันทีโดยไม่ต้องมีการคอมไพล์ไปเป็นภาษาเครื่องอีก เนื่องจากเป็นคำสั่งที่ระบบปฏิบัติการรู้จักอยู่แล้ว

## 2.6 เครื่องมือที่ใช้

### 2.6.1 คอมพิวเตอร์โน้ตบุ๊ก 1 เครื่อง

ใช้ในการเขียนโปรแกรมทุกอย่าง ทั้งการสร้างฐานข้อมูล การโหลดข้อมูลเข้าฐานข้อมูล การสร้างรายงานในโปรแกรม Power BI และอื่น ๆ

### 2.6.2 Window 10

วินโดวส์ คือระบบปฏิบัติการคอมพิวเตอร์ (OS: Operating System) ถูกสร้างขึ้นโดยบริษัทไมโครซอฟต์ (Microsoft) เพราะการใช้งานดอส (DOS: Disk Operating System) ที่มีระบบการทำงานโดยใช้คำสั่งที่ละบรรทัดมีความยุ่งยากทำให้มีการพัฒนาซอฟต์แวร์ที่เรียกว่าวินโดวส์ขึ้นมา ซึ่งมีลักษณะเป็นจียูไอ (GUI: Graphic-User Interface) คือการนำรูปแบบสัญลักษณ์ภาพกราฟิกแทนการใส่คำสั่งที่ละบรรทัดเพื่อให้ใช้งานได้ง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



## 2.6.3 Microsoft Excel

เป็นโปรแกรมทางด้านตารางคำนวณ หรือที่เรียกว่า เสปรดชีต (Spreadsheet) เป็นโปรแกรมในชุด Microsoft Office มีความสามารถในการสร้างตาราง การคำนวณ การวิเคราะห์ การออกรายงานในรูปแบบตารางและกราฟ



ภาพที่ 2.4 ภาพแสดงสัญลักษณ์ของโปรแกรม Microsoft Excel

## 2.6.4 Jupyter Notebook 5.1

เป็นหน้าเว็บที่ประกอบด้วย ช่อง ๆ cell เรียงต่อกันลงไป โดยแต่ละ cell สามารถเป็นเนื้อหา static content ต่าง ๆ เช่น ข้อความ รูปภาพ กราฟ วิดีโอ เสียง หรือ เป็นโค้ดโปรแกรมคอมพิวเตอร์ ภาษา Python ที่สามารถรันคำสั่งประมวลผล แสดงผลลัพธ์ออกมาได้ โดยสามารถรันโค้ดทีละ Cell ได้ โดยมี Library หลักที่ใช้ในการทำรายงานงบบัญชีไรชาดทุน ได้แก่ Pandas, Numpy, sqlalchemy, pathlib, csv, glob

A screenshot of a Jupyter Notebook interface. The top bar shows 'Jupyter Load HO&DC to MSSQL Last checked: 2022-08-08 10:00:00'. The notebook contains several code cells. The first cell imports libraries: pandas, numpy, time, sqlalchemy, and os.path. The second cell defines a function 'init\_mssql\_connection()' that connects to a Microsoft SQL Server. The third cell lists files in a directory. The fourth cell shows the output of the file listing.

```
In [15]: import pandas as pd
import numpy as np
import time

import sqlalchemy as sa
from sqlalchemy import create_engine
from sqlalchemy import event
from os import listdir
from os.path import isfile, join

In [16]: def init_mssql_connection():
server = 'THTVDRASQLH001'
database = 'FIN'

print('*****MS SQL Server Settings*****')
print('Server: ', server)
print('Database: ', database)
conn_string = 'mssql+pyodbc://(//)@%s/%s?Trusted_Connection=yes' % (server, database)
print(conn_string)
engine = sa.create_engine(conn_string, encoding='utf8')

return engine
engine = init_mssql_connection()
*****MS SQL Server Settings*****
Server: THTVDRASQLH001
Database: FIN
mssql+pyodbc://THTVDRASQLH001/FIN?driver=SQL Server?Trusted_Connection=yes

In [17]: filename = [f for f in listdir("C:\\Users\\TH390221325\\OneDrive - Tesco\\Documents\\FP&I\\input\\Actual") if isfile(join(r"C:\\Users\\", filename))]

In [18]: filename
```

ภาพที่ 2.5 ภาพแสดงหน้าต่างของ Jupyter Notebook

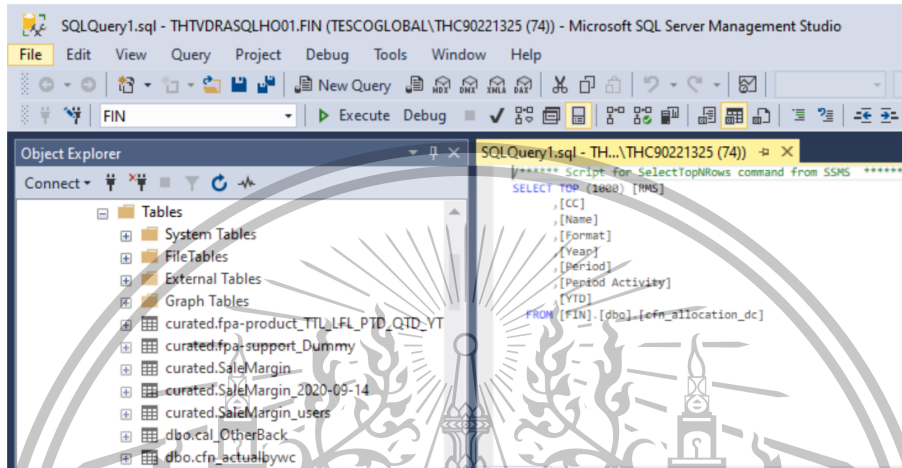
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## 2.6.5 Microsoft SQL server management studio 2018

เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (RDBMS: Relational Database Management System) ที่ถูกพัฒนาขึ้นมาโดยบริษัทไมโครซอฟท์ เป็นซอฟต์แวร์ที่มีหน้าที่หลักในการจัดเก็บข้อมูลและเรียกข้อมูลโดยซอฟต์แวร์อื่น ๆ ซึ่งอาจทำงานบนคอมพิวเตอร์เครื่องเดียวกันหรือบนคอมพิวเตอร์เครื่องอื่นในเครือข่าย



ภาพที่ 2.6 ภาพแสดงหน้าต่างของโปรแกรม Microsoft SQL server management studio

ภาษาเอสคิวแอล (SQL: Structured Query Language) เป็นภาษาที่ใช้ในการเขียนโปรแกรมเพื่อจัดการฐานข้อมูล ถูกสร้างขึ้นมาโดยบริษัท IBM ซึ่งเป็นภาษาที่เป็นระบบเปิด หมายถึงสามารถนำคำสั่งเอสคิวแอลต่าง ๆ ไปใช้กับฐานข้อมูลชนิดใดก็ได้ และคำสั่งเดียวกันเมื่อส่งผ่านระบบฐานข้อมูลที่แตกต่างกัน จะได้ผลลัพธ์ที่เหมือนกัน ผู้ใช้งานจึงสามารถเลือกใช้ฐานข้อมูลชนิดไหนก็ได้

เอสคิวแอลมีโครงสร้างของภาษาที่เข้าใจง่าย ไม่ซับซ้อน และมีประสิทธิภาพสูง เอสคิวแอลจึงเหมาะสำหรับใช้กับระบบฐานข้อมูลเชิงสัมพันธ์ โดยเอสคิวแอลจะแบ่งการทำงานได้ 4 ประเภทดังนี้

- 1) Select query ใช้ในการดึงข้อมูล
- 2) Update query ใช้ในการแก้ไขข้อมูล
- 3) Insert query ใช้ในการเพิ่มข้อมูล
- 4) Delete query ใช้ในการลบข้อมูล

ประโยชน์ของภาษาเอสคิวแอลมีดังนี้

- 1) สามารถสร้างฐานข้อมูลและตาราง
- 2) สนับสนุนการจัดการฐานข้อมูล ประกอบไปด้วยการเพิ่มข้อมูล การปรับปรุงข้อมูล และการลบข้อมูล
- 3) สนับสนุนการค้นหาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

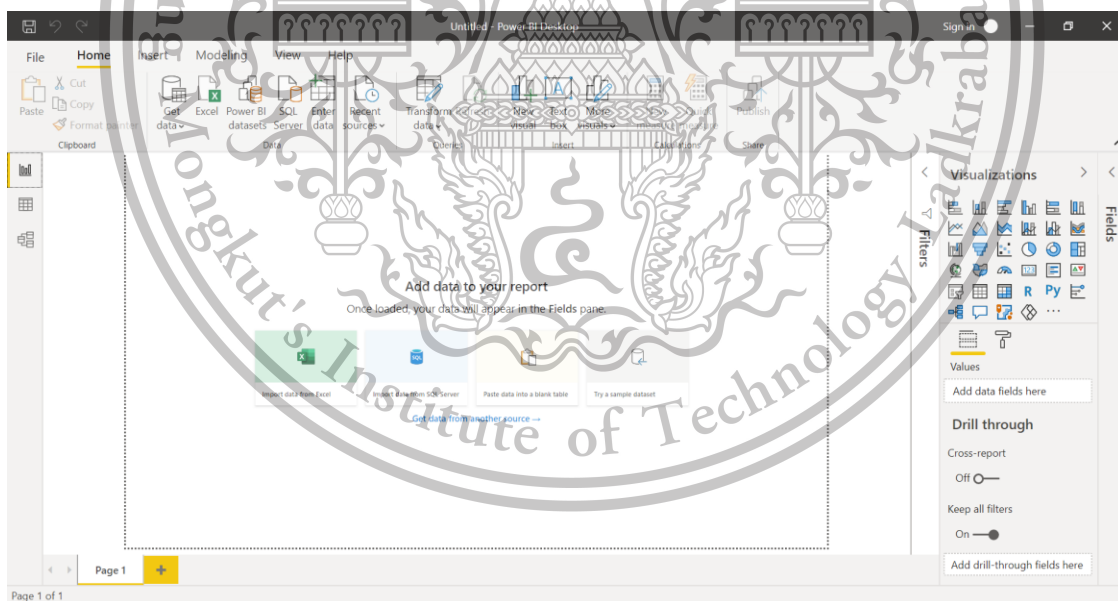


ประเภทของคำสั่งภาษาเอสคิวแอล มีดังนี้

- 1) ภาษานิยามข้อมูล (Data Definition Language: DDL) เป็นคำสั่งที่ใช้ในการสร้างฐานข้อมูล กำหนดโครงสร้างข้อมูล เช่น คำสั่ง DELTE, DROP, ALTER
- 2) ภาษาจัดการข้อมูล (Data Manipulation Language: DML) เป็นคำสั่งที่ใช้ในการเรียกใช้ เพิ่ม ลบ และเปลี่ยนแปลงข้อมูล เช่น คำสั่ง SELECT, INSERT, UPDATE
- 3) ภาษาควบคุมข้อมูล (Data Control Language: DCL) เป็นคำสั่งที่ใช้ในการกำหนดสิทธิการอนุญาต หรือยกเลิกการเข้าถึงฐานข้อมูลเพื่อความปลอดภัยของฐานข้อมูล เช่น คำสั่ง GRANT, REVOKE

### 2.6.6 Microsoft Power BI Desktop

เป็นโปรแกรมที่ใช้ช่วย วิเคราะห์ สรุปผลข้อมูลจำนวนไม่จำกัด จากหลาย ๆ แหล่งข้อมูล ไม่ว่าจะเป็น Excel File, Microsoft Access Database, SQL Server, Oracle เป็นต้น ได้อย่างรวดเร็ว สามารถแสดงผลได้ทั้งรูปแบบ ตาราง สรุปผลด้วย Visualization กราฟในรูปแบบต่าง ๆ และยังติดตั้งเพิ่มเติมได้จาก Marketplace และไฮไลต์ก็คือ แสดงผลแบบเป็นแผนที่ได้อีกด้วย โดยสามารถแสดงผลผ่านเว็บไซต์ และอุปกรณ์ Mobile และ Tablet ได้อีกด้วยทำให้เรา ได้ข้อมูลที่ถูกต้องและรวดเร็ว พร้อมในการกำหนดกลยุทธ์ และตัดสินใจได้อย่างถูกต้องและแม่นยำในทางธุรกิจ



ภาพที่ 2.7 ภาพแสดงหน้าต่างของ Microsoft Power BI

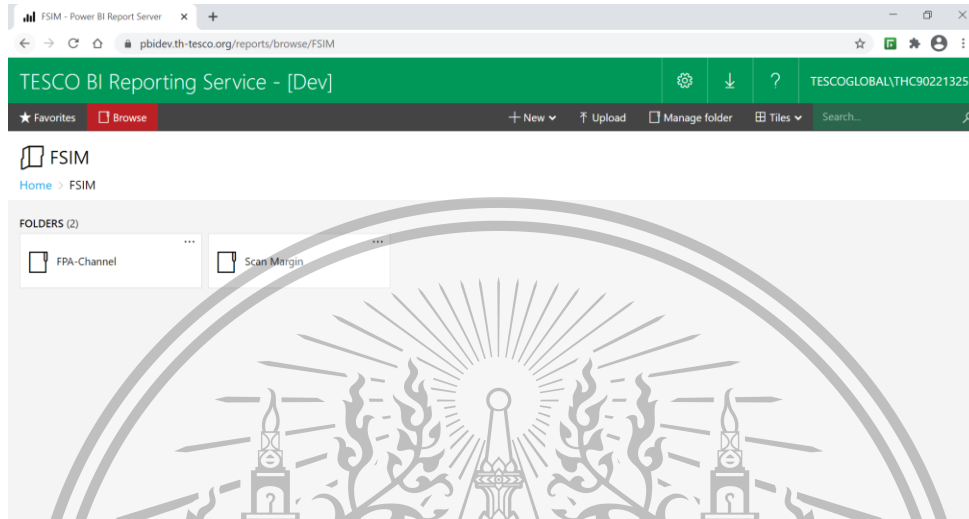
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## 2.6.7 Microsoft Power BI Report Server

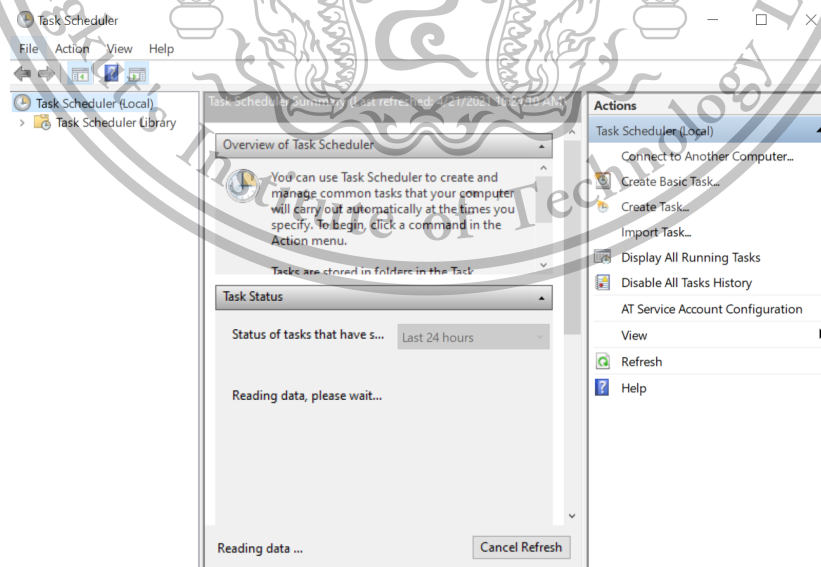
การตั้ง Server ที่เป็น On Premise ให้เป็น Power BI Portal ที่สามารถ บริหารจัดการรายงานแล้ว ควบคุมสิทธิการเข้าถึงของ User โดยตัว Report Server จะทำหน้าที่แทน Power BI Portal ที่อยู่บน Cloud ได้



ภาพที่ 2.8 ภาพแสดงหน้าต่างของ Microsoft Power BI Server

## 2.6.8 Task Scheduler

เป็นโปรแกรมที่มีอยู่ใน Windows ทำหน้าที่ในการตั้งเวลาให้โปรแกรมทำงานเมื่อถึงเวลาที่กำหนดหรือ เกิดเหตุการณ์อะไรบางอย่างขึ้น ตามที่ได้ตั้งค่าไว้ ตัว Scheduler จะทำการเรียก Scheduler นั้น ๆ เพื่อไปเรียก โปรแกรมที่ตั้งค่าเอาไว้ให้ทำงาน



ภาพที่ 2.9 ภาพแสดงหน้าต่างของ Task Scheduler

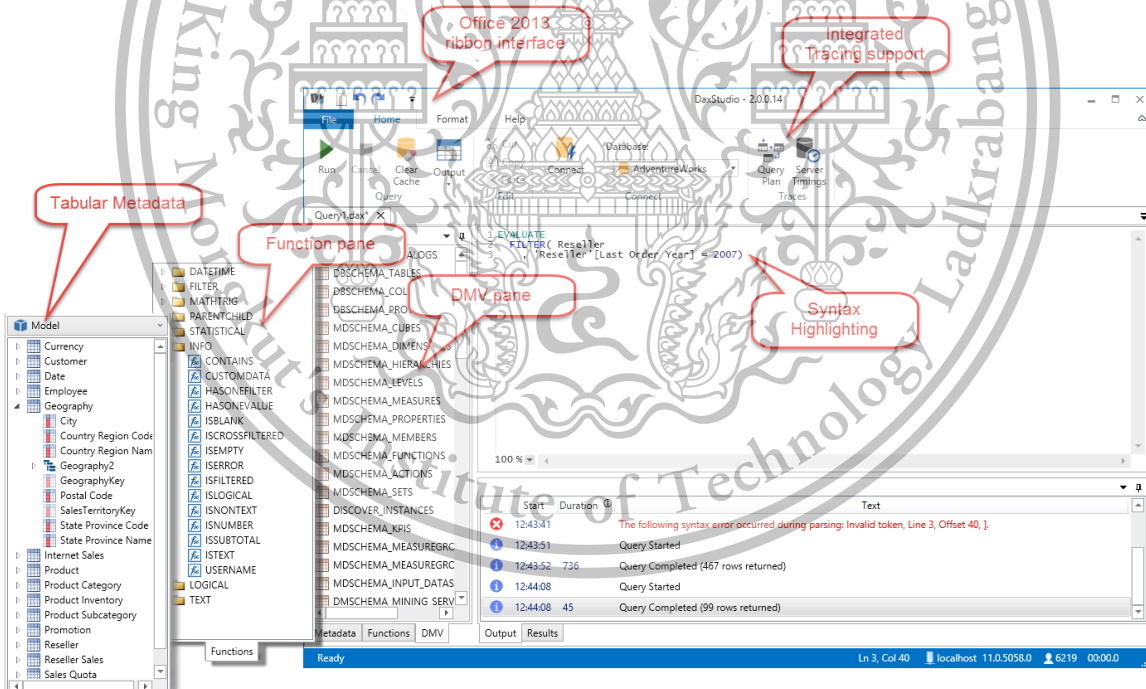
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนเพื่อจุดประสงค์เฉพาะเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6.9 Dax Studio

Dax Studio เป็นเครื่องมือที่ในการจัดการกับภาษา Dax ใช้งานร่วมกับโปรแกรมอื่นอย่างเช่น

- PowerPivot in Excel
- Power BI Desktop
- SSDT Integrated Workspaces
- Analysis Services Tabular
- Azure Analysis Services

DAX คือชุดของ Functions, Operations และ Constants เพื่อการคำนวณใน Data Model ย่อมาจาก Data Analysis Expression โดย DAX สามารถนำไปใช้ใน Power BI หรือ Excel (Power Pivot) หรือ SQL Server ได้ เพื่อคำนวณสร้าง Measure, Column, Table โดยมีสูตรต่าง ๆ มากมาย มีประโยชน์มากในการทำงานเกี่ยวกับ Time Intelligence การสร้าง Date Dimension การคำนวณโดยใช้ Relationships เพื่อเตรียมการคำนวณให้ไปใช้ในการสร้าง Visualization ต่อไป



ภาพที่ 2.10 ภาพแสดงหน้าต่างของ Dax Studio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## บทที่ 3

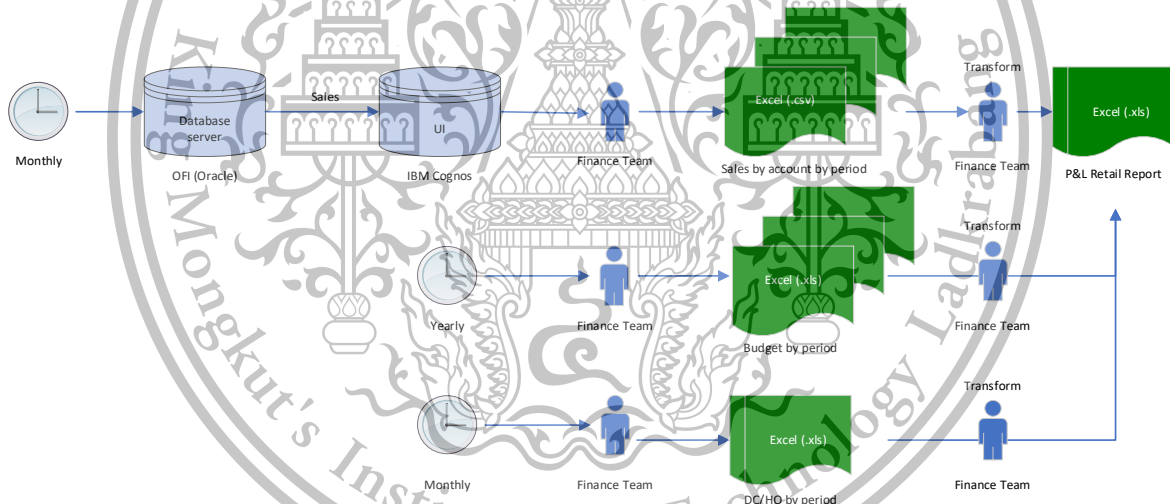
### วิธีการดำเนินการ

#### 3.1 ศึกษารายงานงบกำไรขาดทุนที่ใช้ในปัจจุบัน

##### 3.1.1 กระบวนการจัดทำรายงานปัจจุบัน

ปัจจุบันการจัดทำงบกำไรขาดทุนของบริษัทจะจัดทำขึ้นเดือนละครั้ง ภายในเวลาหนึ่งอาทิตย์หลังจากปิดบัญชี เมื่อจัดทำเสร็จสมบูรณ์แล้วจะมีการส่งให้กับผู้บริหารและร้านค้าปลีกของบริษัทจำนวนประมาณ 2000 สาขาทั่วประเทศ โดยในขั้นตอนการจัดทำรายงานงบกำไรขาดทุนจะใช้ทรัพยากรของพนักงานจากฝ่ายการเงินทั้งหมดสามคน ดำเนินการจัดทำรายงานงบกำไรขาดทุนเป็นเวลาสามวัน (เวลาทำงาน 8 ชั่วโมง) รวมใช้เวลาในการจัดทำรายงานงบกำไรขาดทุนถึง 48 ชั่วโมง ต่อเดือน

รายงานงบกำไรขาดทุนถูกจัดทำโดยใช้โปรแกรม Microsoft Excel เริ่มจากนำข้อมูลทั้งหมดที่จำเป็นและต้องการแสดงบนรายงาน คัดลอกข้อมูลลงในไฟล์ Excel จากนั้นใส่สูตรต่าง ๆ คำนวณข้อมูล เชื่อมโยงข้อมูลกลุ่มหนึ่งกับข้อมูลกลุ่มอื่น ๆ และทำการจัดรูปแบบหน้าตาของข้อมูล



ภาพที่ 3.1 ภาพแสดงกระบวนการจัดทำรายงานปัจจุบัน

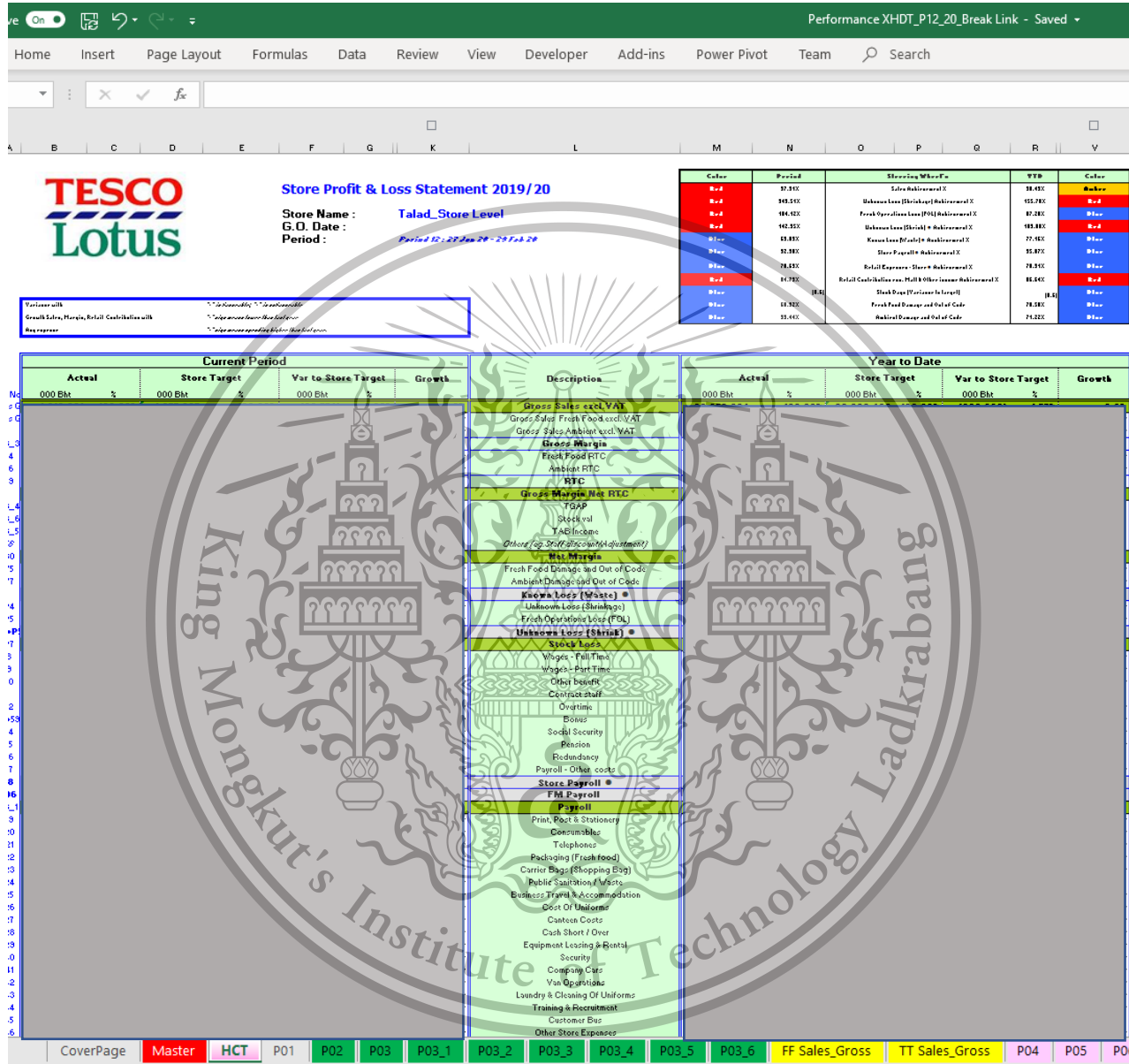
ทางบริษัทจัดทำรายงานงบกำไรขาดทุนออกมาเพื่อใช้ดูข้อมูลในมุมมองของรายได้ ค่าใช้จ่าย ที่เกิดขึ้นในแต่ละเดือน และดูเปรียบเทียบข้อมูลรายได้ ค่าใช้จ่ายแต่ละเดือนนั้น ๆ กับเป้าหมายที่ตั้งเป้าไว้ว่าจะเกิดขึ้น และเปรียบเทียบกับรายได้ ค่าใช้จ่ายที่เกิดขึ้นของเดือนนั้น ๆ เมื่อปีที่แล้ว ดูเปอร์เซ็นต์การเติบโตของยอดขาย เพื่อให้เห็นผลการดำเนินการของกิจการได้อย่างชัดเจน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ทางฝ่ายการเงินจะทำการส่งรายงานงบกำไรขาดทุนนี้ไปยังผู้จัดการของรายค้าปลีกประมาณ 2000 สาขาทั่วประเทศ ทางผู้จัดการร้านค้าได้นำไปใช้ในการจัดการบริหารร้านค้าต่อไป เพื่อให้การดำเนินธุรกิจเป็นไปอย่างราบรื่น ผู้จัดการร้านค้าปลีกเห็นภาพรวมของร้านค้าที่ตนบริหารอยู่ สามารถปรับเปลี่ยนบริหารเงินได้ดียิ่งขึ้น



ภาพที่ 3.2 ภาพแสดงหน้ารายงานงบกำไรขาดทุนที่ใช้ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



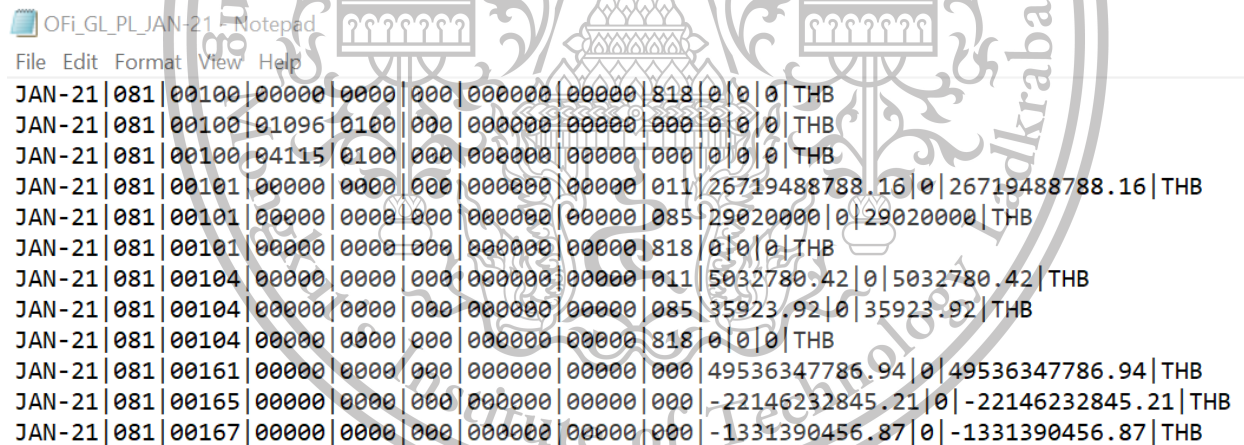
### 3.1.2 ประเภทของรายงานงบกำไรขาดทุน

รายงานงบกำไรขาดทุนของบริษัทโลตัส ของทีม Financial Process and Analyst – Channel แบ่งออกเป็น 2 แบบ ได้แก่

- รายงานงบกำไรขาดทุน แบบ Full set คือ มีการกระจายค่าใช้จ่ายส่วนกลาง ค่าใช้จ่ายที่ไม่สามารถระบุได้ว่าเป็นค่าใช้จ่ายของร้านค้าขายปลีกสาขาไหน กระจายค่าใช้จ่ายตรงส่วนนี้ให้กับทุกสาขาแยกตามประเภทของร้านค้าที่แบ่งออกเป็น Hypermarket, Department Store, Extra, Talad, Express โดยจะกระจายค่าใช้จ่ายตามเปอร์เซ็นต์ยอดขายของแต่ละสาขา
- รายงานงบกำไรขาดทุน แบบ KPI คือ ไม่คิดในส่วนของค่าใช้จ่ายส่วนกลาง ค่าใช้จ่ายที่ไม่สามารถระบุได้ว่าเป็นค่าใช้จ่ายของร้านค้าขายปลีกสาขาไหน คิด วิเคราะห์แค่เฉพาะรายได้และค่าใช้จ่ายที่ระบุได้ของแต่ละสาขาเท่านั้น

### 3.1.3 ข้อมูลที่นำมาใช้งาน

- ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นของแต่ละร้านค้าปลีก (Trial balance) เป็นข้อมูลรายได้ ค่าใช้จ่ายที่เกิดขึ้นในเดือนนั้น ๆ ได้รับข้อมูลมาจากฝ่ายดูแลระบบฐานข้อมูลของแผนกการเงิน โดยไฟล์ที่ได้มาเป็นไฟล์ text จำนวน 1 ไฟล์ คำนข้อมูลด้วยสัญลักษณ์ “|” ไม่มีชื่อคอลัมน์



Month	Store Code	Revenue	Expenses	Net Income	Other	Revenue	Expenses	Net Income	Other	Revenue	Expenses	Net Income	Other	Revenue	Expenses	Net Income	Other	
JAN-21	081	00100	00000	0000	000	000000	00000	818	0	0	0	THB						
JAN-21	081	00100	01096	0100	000	000000	00000	000	0	0	0	THB						
JAN-21	081	00100	04115	0100	000	000000	00000	000	0	0	0	THB						
JAN-21	081	00101	00000	0000	000	000000	00000	011	26719488788.16	0	26719488788.16	THB						
JAN-21	081	00101	00000	0000	000	000000	00000	085	29020000	0	29020000	THB						
JAN-21	081	00101	00000	0000	000	000000	00000	818	0	0	0	THB						
JAN-21	081	00104	00000	0000	000	000000	00000	011	5032780.42	0	5032780.42	THB						
JAN-21	081	00104	00000	0000	000	000000	00000	085	35923.92	0	35923.92	THB						
JAN-21	081	00104	00000	0000	000	000000	00000	818	0	0	0	THB						
JAN-21	081	00161	00000	0000	000	000000	00000	000	49536347786.94	0	49536347786.94	THB						
JAN-21	081	00165	00000	0000	000	000000	00000	000	-22146232845.21	0	-22146232845.21	THB						
JAN-21	081	00167	00000	0000	000	000000	00000	000	-1331390456.87	0	-1331390456.87	THB						

ภาพที่ 3.3 ภาพแสดงข้อมูล Trial balance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- ข้อมูล Budget เป็นข้อมูลที่ทางทีม FPA-Channel ทำขึ้นเพื่อเป็นการตั้งเป้ายอดขาย ค่าใช้จ่ายให้กับร้านค้าปลีกประมาณ 2000 ร้านทั่วประเทศ จัดทำขึ้นปีละครั้งเท่านั้น โดยไฟล์ข้อมูลที่ได้มาเป็นไฟล์ Excel ประมาณ 20 ไฟล์ และในหนึ่งไฟล์มีหลายชีต

				01-Mar-20	06-Apr-20	04-May-20	01-Jun-20	06-Jul-20	03-Aug-20	31-Aug-20	05-Oct-20	02-Nov-20	30-Nov-20	04-Jan-21	01-Feb-21	
				P01	P02	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	
Account	Cost	Cost Type	Prod_Local													
36100	41500	0141	000	00401												
36100	43802	0141	000	00401	1,600.97	1,201.35	1,327.51	1,612.73	1,386.50	1,435.03	2,007.80	1,656.77	1,779.04	1,662.71	1,657.23	1,647.79
36100	43803	0141	000	00401	1,612.34	1,209.83	1,336.83	1,623.98	1,396.13	1,444.95	2,021.59	1,670.11	1,790.68	1,675.25	1,649.03	1,642.21
36100	43805	0141	000	00401	1,605.15	1,204.46	1,330.92	1,616.84	1,390.01	1,438.64	2,012.80	1,662.88	1,783.42	1,621.18	1,649.29	1,642.21
36100	43806	0141	000	00401	1,606.31	1,205.34	1,331.89	1,618.01	1,391.02	1,439.68	2,014.27	1,664.09	1,784.72	1,621.26	1,649.29	1,642.21
36100	43807	0141	000	00401	1,633.51	1,225.63	1,354.21	1,644.99	1,414.09	1,463.46	2,047.36	1,607.48	1,604.05	1,636.58	1,631.67	1,522.73
36100	43808	0141	000	00401	1,603.83	1,203.48	1,329.85	1,615.55	1,388.92	1,437.52	2,011.26	1,661.62	1,782.08	1,632.16	1,651.94	1,640.46
36100	43809	0141	000	00401	1,500.57	1,125.85	1,243.93	1,510.99	1,298.86	1,344.17	1,864.57	1,812.99	1,911.57	1,982.88	1,824.44	1,842.51
36100	43810	0141	000	00401	1,492.90	1,120.11	1,237.69	1,503.30	1,292.27	1,337.36	1,952.89	2,084.73	2,272.45	2,162.40	2,036.42	2,019.29
36100	43811	0141	000	00401	1,161.48	871.47	962.89	1,160.65	1,005.47	1,040.57	1,455.75	1,118.84	1,172.89	1,169.43	1,098.92	1,090.54
36100	43812	0141	000	00401	1,440.24	1,080.62	1,193.99	1,450.35	1,246.78	1,290.31	1,805.12	1,739.96	2,236.57	1,609.97	1,864.36	1,851.38
36100	43813	0141	000	00401	1,719.00	1,289.77	1,428.05	1,731.08	1,488.09	1,540.04	2,154.50	1,936.71	2,003.26	2,003.67	1,928.16	1,911.67
36100	43814	0141	000	00401	1,485.37	1,114.49	1,231.42	1,495.85	1,285.89	1,330.79	1,861.78	1,331.20	1,222.62	1,237.46	1,180.72	1,166.62

ภาพที่ 3.4 ภาพแสดงข้อมูล Budget

- ข้อมูลรายได้และค่าใช้จ่ายในส่วนของคุณ้กระจายสินค้า เป็นข้อมูลที่ทางทีม FPA-Channel ทำขึ้นเดือนละครั้ง ข้อมูลแสดงรายได้และค่าใช้จ่ายที่เกิดขึ้นของคุณ้กระจายสินค้าในเดือนนั้น ๆ โดยไฟล์ข้อมูลที่ได้มาเป็นไฟล์ Excel

DC Cost Allocation				3	4	5	6	7	8	9	10	11	12	13
Unit: baht														
				Extra	Hyper	Department store	Talad	Express	All					
				(68,736,114)	(21,223,338)	(17,890,677)	(2,872,243)	(18,616,206)	(23,466,304)	(21,791,900)	(16,772,909)	(16,052,662)	(22,106,723)	(16,834,855)
				(168,716,496)	(130,927,921)	(85,083,858)	(87,828,959)	(55,122,068)	(44,047,881)	(42,461,441)	(33,737,134)	(32,447,597)	(47,384,266)	(33,509,753)
				(15,485,893)	(9,767,411)	(36,083,053)	(73,826,155)	(5,177,058)	(44,037,381)	(31,061,341)	(33,367,134)	(32,447,597)	(47,384,266)	(33,509,753)
				(78,359,752)	(76,794,424)	(59,818,114)	(72,833,632)	(54,848,778)	(71,072,355)	(71,566,252)	(55,623,812)	(52,957,578)	(79,913,929)	(58,094,821)
				(147,353,166)	(147,009,251)	(123,100,241)	(164,289,851)	(109,629,478)	(138,344,843)	(153,865,126)	(121,185,055)	(116,085,116)	(152,135,349)	(120,750,904)
				(472,269,421)	(415,216,921)	(355,533,855)	(466,518,879)	(340,655,455)	(418,505,749)	(430,009,301)	(341,904,880)	(324,882,223)	(455,616,731)	(340,569,948)

ภาพที่ 3.5 ภาพแสดงข้อมูล DC

- ข้อมูลรายได้และค่าใช้จ่ายในส่วนของสำนักงานใหญ่ เป็นข้อมูลที่ทางทีม FPA-Channel ทำขึ้นเดือนละครั้ง ข้อมูลแสดงรายได้และค่าใช้จ่ายที่เกิดขึ้นของสำนักงานใหญ่ในเดือนนั้น ๆ โดยไฟล์ข้อมูลที่ได้มาเป็นไฟล์ Excel

DC Cost Allocation				3	4	5	6	7	8	9	10	11	12	13
Unit: baht														
				Extra	Hyper	Department store	Talad	Express	All					
				(26,735,114)	(21,223,338)	(17,890,677)	(2,872,243)	(18,616,206)	(23,466,304)	(21,791,900)	(16,772,909)	(16,052,662)	(22,106,723)	(16,834,855)
				(168,716,496)	(130,927,921)	(85,083,858)	(87,828,959)	(55,122,068)	(44,047,881)	(42,461,441)	(33,737,134)	(32,447,597)	(47,384,266)	(33,509,753)
				(15,485,893)	(9,767,411)	(36,083,053)	(73,826,155)	(5,177,058)	(44,037,381)	(31,061,341)	(33,367,134)	(32,447,597)	(47,384,266)	(33,509,753)
				(78,359,752)	(76,794,424)	(59,818,114)	(72,833,632)	(54,848,778)	(71,072,355)	(71,566,252)	(55,623,812)	(52,957,578)	(79,913,929)	(58,094,821)
				(147,353,166)	(147,009,251)	(123,100,241)	(164,289,851)	(109,629,478)	(138,344,843)	(153,865,126)	(121,185,055)	(116,085,116)	(152,135,349)	(120,750,904)
				(472,269,421)	(415,216,921)	(355,533,855)	(466,518,879)	(340,655,455)	(418,505,749)	(430,009,301)	(341,904,880)	(324,882,223)	(455,616,731)	(340,569,948)

ภาพที่ 3.6 ภาพแสดงข้อมูล HO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational 16e only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

### 3.1.4 รายการประเภทของรายได้และค่าใช้จ่าย

- 1) Net sale excludes VAT คือ รายได้สุทธิของสินค้าทั้งหมดที่ไม่รวมภาษีมูลค่าเพิ่ม
  - Net sales ambient exclude VAT คือ รายได้สุทธิของสินค้าที่ไม่ใช่อาหารสดที่ไม่รวมภาษีมูลค่าเพิ่ม
  - Net sales fresh food excludes VAT คือ รายได้สุทธิของสินค้าของอาหารสดที่ไม่รวมภาษีมูลค่าเพิ่ม
- 2) Net Margin คือ กำไรสุทธิของสินค้าทั้งหมด
  - Gross Margin คือ กำไรสุทธิเบื้องต้นของสินค้า
  - TGAP คือ กำไรที่ได้จากการขายสินค้าประเภทสินค้าจัดชุดเซต
  - Fresh Food RTC คือ ส่วนที่ขาดทุนจากการลดราคาสินค้าเพื่อกำจัดสินค้าอาหารสดที่ขายไม่ได้ออก
  - Ambient RTC คือ ส่วนที่ขาดทุนจากการลดราคาสินค้าเพื่อกำจัดสินค้าไม่ใช่อาหารสดที่ขายไม่ได้ออก
  - Stock Valuation คือ ราคาสินค้าที่มีเก็บอยู่ในคลังสินค้าของร้าน
  - Commercial support – Nielson คือ ค่าใช้จ่ายในการทำโฆษณา
  - Tab income คือ ค่าใช้จ่ายในการทำการตลาด
- 3) Mall and Retail Service คือ รายได้รวมที่เกิดจาก Mall และ Retail service
  - Mall คือ กำไรที่เกิดขึ้นในส่วนของ Mall
  - Retail service คือ กำไรที่เกิดขึ้นในส่วนของ Retail service
- 4) Centrally cost คือ ค่าใช้จ่ายรวมที่เกิดจากศูนย์กลางต่าง ๆ
  - Net distribution center คือ ค่าใช้จ่ายที่เกิดขึ้นในส่วนของคุณยกระจ่ายสินค้า
  - Net head office คือ ค่าใช้จ่ายที่เกิดขึ้นในส่วนของสำนักงานใหญ่
  - Net marketing คือ ค่าใช้จ่ายที่เกิดขึ้นในส่วนของการทำการตลาด
- 5) Stock loss คือ ค่าใช้จ่ายที่เกิดจากการที่สินค้าไม่สามารถนำไปขายได้
  - Known loss (Waste) คือ ค่าใช้จ่ายที่เกิดจากสินค้าสูญหาย โดยสามารถระบุได้ว่าสินค้าสูญหายอย่างไร เช่น สินค้าหมดอายุ สินค้าชำรุด
  - Unknown loss (Shrink) คือ ค่าใช้จ่ายที่เกิดจากสินค้าสูญหาย โดยไม่สามารถระบุได้ว่าสินค้าสูญหายอย่างไร เช่น สินค้าโดนขโมย ผลไม้ที่ลูกค้ำชิม
- 6) Property cost คือ ค่าใช้จ่ายที่เกิดจากพื้นที่ขายสินค้า
  - Property rent and others คือ ค่าใช้จ่ายที่เกิดจากการเช่าพื้นที่ขายสินค้า
  - Depreciation property คือ ค่าใช้จ่ายที่เกิดจากการเสื่อมราคาของสินทรัพย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

7) Store expenses คือ ค่าใช้จ่ายที่เกิดขึ้นในร้านค้า

- Retail FM expenses คือ ค่าใช้จ่ายที่เกิดขึ้นในร้านค้าปลีก
- Bank change คือ ค่าใช้จ่ายเกี่ยวกับธนาคาร
- Retail expenses – store คือ ค่าใช้จ่ายของร้านค้าปลีก
- Depreciation non – property คือ ค่าใช้จ่ายที่เกิดจากการเสื่อมราคาของสินทรัพย์และทรัพย์สินภายในร้านค้า
- FM excludes payroll คือ ค่าใช้จ่ายจำพวกของใช้ทำอาหาร ค่าน้ำ ค่าไฟ และอื่น ๆ

8) Payroll คือ ค่าใช้จ่ายในส่วนของเงินเดือนพนักงาน

- FM payroll คือ ค่าใช้จ่ายในส่วนของเงินเดือนพนักงานประจำร้านค้า
- Store payroll คือ ค่าใช้จ่ายในส่วนของเงินเดือนพนักงานชั่วคราวร้านค้า

### 3.2 เก็บรวบรวมสิ่งที่ผู้ใช้งานต้องการ และปัญหาที่ผู้ใช้งานประสบ

ในด้านของการจัดทำรายงานงบกำไรขาดทุน ปัญหาที่ผู้จัดทำ (ฝ่ายการเงิน) พบเจอ คือ ปัญหาเรื่องเวลา เนื่องจากข้อมูลที่นำมาจากฐานข้อมูลไม่สามารถนำมาใช้ได้ทันที ต้องมีการเพิ่มสูตร การทำให้ข้อมูลอยู่ในมุมมองที่ต้องการ จึงทำให้ต้องใช้เวลาในการจัดทำถึง 48 ชั่วโมง ต่อเดือน และปัญหาเรื่องความถูกต้องแม่นยำ เนื่องจากรายงานงบกำไรขาดทุนถูกจัดทำโดยพนักงานฝ่ายการเงินในทุกขั้นตอน ในบางครั้งจึงเกิดข้อผิดพลาดที่เกิดจากมนุษย์ขึ้น ทำให้ข้อมูลผิดเพี้ยนไปจากความเป็นจริง

ส่วนในด้านของการใช้งานรายงานงบกำไรขาดทุน ปัญหาที่ผู้ใช้งานพบเจอ คือ ปัญหาเรื่องข้อจำกัดของโปรแกรม Microsoft Excel เนื่องจากข้อมูลที่นำมาใช้ทำรายงานงบกำไรขาดทุนนั้นเป็นข้อมูลขนาดใหญ่ จึงต้องมีการแบ่งรายงานงบกำไรขาดทุนออกเป็น 2 ไฟล์ จึงจะสามารถใช้งานได้

### 3.3 คิด วิเคราะห์ หาทางแก้ไขปัญหา

แก้ปัญหาโดยการจัดทำรายงานให้เป็นแบบอัตโนมัติ ตั้งแต่การนำข้อมูลเข้าฐานข้อมูล การเพิ่มสูตรคำนวณข้อมูล การเตรียมข้อมูลให้อยู่ในรูปแบบ มุมมองและระดับที่ต้องการ สูตรในการคำนวณที่ต้องเพิ่มเติมนั้น ไม่มีการเปลี่ยนแปลง ดังนั้นการใช้โปรแกรมในการคำนวณทำให้สามารถมั่นใจได้ว่าข้อมูลที่ได้ออกต้อง ลดความผิดพลาดที่เกิดจากมนุษย์

จากข้อจำกัดของโปรแกรม Microsoft Excel จึงเปลี่ยนมาใช้โปรแกรม Microsoft Power BI แทน ซึ่งสามารถรองรับข้อมูลได้จำนวนมาก และมีฟังก์ชันที่เอื้อต่อการวิเคราะห์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

### 3.4 ออกแบบและพัฒนา

#### 3.4.1 ออกแบบฐานข้อมูล เพื่อใช้ในการเก็บข้อมูล

ออกแบบฐานข้อมูล โครงสร้างของแต่ละตารางที่ใช้เก็บข้อมูล มี 6 ตาราง ได้แก่

1) ออกแบบตารางที่ใช้ในการเก็บข้อมูลรายได้หรือค่าใช้จ่ายที่เกิดขึ้นจริง (Actual data) โดยโครงสร้างของตารางเป็นดังนี้

- คอลัมน์ที่ 1 คือ คอลัมน์ Group ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 2 คือ คอลัมน์ Acc ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 3 คือ คอลัมน์ CG ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 4 คือ คอลัมน์ CCT ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 5 คือ คอลัมน์ PRD ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 6 คือ คอลัมน์ PRJ ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 7 คือ คอลัมน์ Local ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 8 คือ คอลัมน์ Inter ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 9 คือ คอลัมน์ Year ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 10 คือ คอลัมน์ Period ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 11 คือ คอลัมน์ Beginning Balance ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 12 คือ คอลัมน์ Period Activity ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูล

เอกสารนี้เป็นวงจำกัดที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- คอลัมน์ที่ 13 คือ คอลัมน์ Ending Balance ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้

คอลัมน์ที่กำหนดให้เป็นคีย์ของตารางได้แก่คอลัมน์ Group, Acc, CC, CCT, PRD, PRJ, Local, Inter, Year, Period

Column Name	Data Type	Allow Nulls
[Group]	varchar(10)	<input type="checkbox"/>
Acc	varchar(10)	<input type="checkbox"/>
CC	varchar(10)	<input type="checkbox"/>
CCT	varchar(10)	<input type="checkbox"/>
PRD	varchar(10)	<input type="checkbox"/>
PRJ	varchar(10)	<input type="checkbox"/>
Local	varchar(10)	<input type="checkbox"/>
Inter	varchar(10)	<input type="checkbox"/>
Year	varchar(5)	<input type="checkbox"/>
Period	varchar(5)	<input type="checkbox"/>
[Beginning Balance]	float	<input checked="" type="checkbox"/>
[Period Activity]	float	<input checked="" type="checkbox"/>
[Ending Balance]	float	<input checked="" type="checkbox"/>

ภาพที่ 3.7 ภาพแสดงโครงสร้างตาราง Actual data

```

1 /***** Script for SelectTopNRows command from SSMS *****/
2 SELECT [Group]
3        ,[Acc]
4        ,[CC]
5        ,[CCT]
6        ,[PRD]
7        ,[PRJ]
8        ,[Local]
9        ,[Inter]
10       ,[Year]
11       ,[Period]
12       ,[Beginning Balance]
13       ,[Period Activity]
14       ,[Ending Balance]
15 FROM [FIN].[dbo].[cfn_trial_balance_detail A]
  
```

	Group	Acc	CC	CCT	PRD	PRJ	Local	Inter	Year	Period	Beginning Balance	Period Activity	Ending Balance
1	081	00201	00000	0000	000	000000	00000	000	2021	04	4999800	-4999800	0
2	081	00910	00000	0000	000	000000	00000	000	2021	04	-5137500050	5137500000	-50
3	081	00910	00000	0000	000	000000	00000	000	2021	05	-50	-5137500000	-5137500050
4	081	00910	00000	0000	000	000000	00000	099	2021	05	-5137500000	5137500000	0
5	081	00976	00000	0000	000	000000	00000	099	2021	05	-2483749997	2483749997	0
6	081	00977	00000	0000	000	000000	00000	000	2020	12	-86181634171.05	513750005	-85667884166.05
7	081	00977	00000	0000	000	000000	00000	000	2021	04	-87891636956.1	2483749997	-85407886959.1
8	081	00977	00000	0000	000	000000	00000	000	2021	05	-85407886959.1	-2483749997	-87891636956.1
9	081	01195	00000	0000	000	000000	00000	000	2020	04	148926939.12	3366729.28	152293668.4
10	081	01195	00000	0000	000	000000	00000	000	2020	12	152293668.4	2859230.34	155152898.74
11	081	01220	00000	0000	000	000000	00000	000	2020	05	-6191055.02	-3577307.46	-9768362.48
12	081	01220	00000	0000	000	000000	00000	000	2020	06	-9768362.48	-2244374.27	-12012736.75
13	081	01220	00000	0000	000	000000	00000	000	2020	07	12012736.75	70000	12082736.75

ภาพที่ 3.8 ภาพแสดงข้อมูล Actual data ในโปรแกรม SSMS

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้ใช้เฉพาะในหน่วยงานราชการเท่านั้น ห้ามนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



2) ออกแบบตารางที่ใช้ในการเก็บข้อมูลรายได้หรือค่าใช้จ่ายที่คาดว่าจะเกิดขึ้น (Target data) โดยโครงสร้างของตารางเป็นดังนี้

- คอลัมน์ที่ 1 คือ คอลัมน์ Group ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 2 คือ คอลัมน์ Acc ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 3 คือ คอลัมน์ CC ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 4 คือ คอลัมน์ CCT ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 5 คือ คอลัมน์ PRD ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 6 คือ คอลัมน์ PRJ ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 7 คือ คอลัมน์ Local ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 8 คือ คอลัมน์ Inter ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 9 คือ คอลัมน์ Year ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 10 คือ คอลัมน์ Period ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 11 คือ คอลัมน์ Period Activity ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 12 คือ คอลัมน์ YTD ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้

คอลัมน์ที่กำหนดให้เป็นคีย์ของตารางได้แก่คอลัมน์ Group, Acc, CC, CCT, PRD, PRJ, Local, Inter, Year, Period

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

	Column Name	Data Type	Allow Nulls
▼	[Group]	varchar(10)	<input type="checkbox"/>
▼	Acc	varchar(10)	<input type="checkbox"/>
▼	CC	varchar(10)	<input type="checkbox"/>
▼	CCT	varchar(10)	<input type="checkbox"/>
▼	PRD	varchar(10)	<input type="checkbox"/>
▼	PRJ	varchar(10)	<input type="checkbox"/>
▼	Local	varchar(10)	<input type="checkbox"/>
▼	Inter	varchar(10)	<input type="checkbox"/>
▼	Year	varchar(5)	<input type="checkbox"/>
▼	Period	varchar(5)	<input type="checkbox"/>
	[Period Activity]	float	<input checked="" type="checkbox"/>
	YTD	float	<input checked="" type="checkbox"/>

ภาพที่ 3.9 ภาพแสดงโครงสร้างตาราง Target data

```

1 /***** Script for SelectTopNRows command from SSMS *****/
2 SELECT [Group]
3       , [Acc]
4       , [CC]
5       , [CCT]
6       , [PRD]
7       , [PRJ]
8       , [Local]
9       , [Inter]
10      , [Year]
11      , [Period]
12      , [Period Activity]
13      , [YTD]
14 FROM [FIN].[dbo].[fpa-channel_budget]

```

88 %

Results Messages

	Group	Acc	CC	CCT	PRD	PRJ	Local	Inter	Year	Period	Period Activity	YTD
16	081	76920	04002	0100	000	000000	000000	00	2021	04	-133592.227001252	-472573.957763697
17	081	76920	04002	0100	000	000000	000000	00	2021	05	106808.370678937	-579382.328442634
18	081	76920	04002	0100	000	000000	000000	00	2021	06	-106808.370678937	-686190.699121571
19	081	76920	04002	0100	000	000000	000000	00	2021	07	-131951.471932359	-818142.17105393
20	081	76920	04002	0100	000	000000	000000	00	2021	08	-105570.529222426	-923712.700276356
21	081	76920	04002	0100	000	000000	000000	00	2021	09	-104330.029818142	-1028042.7300945
22	081	76920	04002	0100	000	000000	000000	00	2021	10	-129829.706491525	-1157872.43658602
23	081	76920	04002	0100	000	000000	000000	00	2021	11	-105647.424042367	-1263519.86062839
24	081	76920	04002	0100	000	000000	000000	00	2021	12	-130100.866707211	-1393620.7273356
25	081	76920	04003	0100	000	000000	000000	00	2021	01	-181685.821494301	-181685.821494301
26	081	76920	04003	0100	000	000000	000000	00	2021	02	-164075.540009269	-345761.36150357
27	081	76920	04003	0100	000	000000	000000	00	2021	03	-164075.540009269	-509836.90151284
28	081	76920	04003	0100	000	000000	000000	00	2021	04	-205094.432253461	-714931.3337663

ภาพที่ 3.10 ภาพแสดงข้อมูล Target data ในโปรแกรม SSMS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ออกแบบตารางที่ใช้ในการเก็บข้อมูลรายได้หรือค่าใช้จ่ายที่เกิดขึ้นจริงของศูนย์กระจายสินค้า (Distribution center data) โดยโครงสร้างของตารางเป็นดังนี้

- คอลัมน์ที่ 1 คือ คอลัมน์ RMS ประเภทของข้อมูลกำหนดให้เป็น varchar 15 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 2 คือ คอลัมน์ CC ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 3 คือ คอลัมน์ Name ประเภทของข้อมูลกำหนดให้เป็น varchar 150 ตัวอักษร สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 4 คือ คอลัมน์ Format ประเภทของข้อมูลกำหนดให้เป็น varchar 50 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 5 คือ คอลัมน์ Year ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 6 คือ คอลัมน์ Period ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 7 คือ คอลัมน์ Period Activity ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 8 คือ คอลัมน์ YTD ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้

คอลัมน์ที่กำหนดให้เป็นคีย์ของตารางได้แก่คอลัมน์ RMS, CC, Format, Year, Period

Column Name	Data Type	Allow Nulls
RMS	varchar(15)	<input type="checkbox"/>
CC	varchar(10)	<input type="checkbox"/>
Name	varchar(150)	<input checked="" type="checkbox"/>
Format	varchar(50)	<input type="checkbox"/>
Year	varchar(5)	<input type="checkbox"/>
Period	varchar(5)	<input type="checkbox"/>
[Period Activity]	float	<input checked="" type="checkbox"/>
YTD	float	<input checked="" type="checkbox"/>

ภาพที่ 3.11 ภาพแสดงโครงสร้างตาราง DC data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```

1 SELECT [RMS]
2      ,[CC]
3      ,[Name]
4      ,[Format]
5      ,[Year]
6      ,[Period]
7      ,[Period Activity]
8      ,[YTD]
9 FROM [FIN].[dbo].[cfn_allocation_dc]

```

	RMS	CC	Name	Format	Year	Period	Period Activity	YTD
1	1	01001	01001 TH-Seacon	Hyper	2020	01	-1442596.22960566	-1442596.22960566
2	1	01001	01001 TH-Seacon	Hyper	2020	02	-1558625.84462323	-3001222.07422889
3	1	01001	01001 TH-Seacon	Hyper	2020	03	-1458041.56336427	-4459263.63759315
4	1	01001	01001 TH-Seacon	Hyper	2020	04	-1697682.10550762	-6156945.74310077
5	1	01001	01001 TH-Seacon	Hyper	2020	05	-1215950.53865117	-7372896.28175194
6	1	01001	01001 TH-Seacon	Hyper	2020	06	-1710219.75391014	-9083116.03566208
7	1	01001	01001 TH-Seacon	Hyper	2020	07	-1938923.32223607	-11022039.3578982
8	1	01001	01001 TH-Seacon	Hyper	2020	08	-1367670.02004564	-12389709.3779438
9	1	01001	01001 TH-Seacon	Hyper	2020	09	-1219064.50169126	-13608773.8796351

ภาพที่ 3.12 ภาพแสดงข้อมูล DC data ในโปรแกรม SSMS

4) ออกแบบตารางที่ใช้ในการเก็บข้อมูลรายได้หรือค่าใช้จ่ายที่เกิดขึ้นจริงของสำนักงานใหญ่ (Head office data) โดยโครงสร้างของตารางเป็นดังนี้

- คอลัมน์ที่ 1 คือ คอลัมน์ Ram ประเภทของข้อมูลกำหนดให้เป็น varchar 15 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 2 คือ คอลัมน์ CC ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 3 คือ คอลัมน์ Name ประเภทของข้อมูลกำหนดให้เป็น varchar 150 ตัวอักษร สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 4 คือ คอลัมน์ Format ประเภทของข้อมูลกำหนดให้เป็น varchar 50 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 5 คือ คอลัมน์ Year ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 6 คือ คอลัมน์ Period ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 7 คือ คอลัมน์ Period Activity ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้

เอกสารนี้เป็น คอลัมน์ที่ 8 คือ คอลัมน์ YTD ประเภทของข้อมูลกำหนดให้เป็น float สามารถเก็บข้อมูลว่างได้ ชนด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



คอลัมน์ที่กำหนดให้เป็นคีย์ของตารางได้แก่คอลัมน์ Ram, CC, Format, Year, Period

Column Name	Data Type	Allow Nulls
Ram	varchar(15)	<input type="checkbox"/>
CC	varchar(10)	<input type="checkbox"/>
Name	varchar(150)	<input checked="" type="checkbox"/>
Format	varchar(50)	<input type="checkbox"/>
Year	varchar(5)	<input type="checkbox"/>
Period	varchar(5)	<input type="checkbox"/>
[Period Activity]	float	<input checked="" type="checkbox"/>
YTD	float	<input checked="" type="checkbox"/>

ภาพที่ 3.13 ภาพแสดงโครงสร้างตาราง HO data

The screenshot shows a SQL query in a query editor and its corresponding results in a grid. The query selects columns Ram, CC, Name, Format, Year, Period, Period Activity, and YTD from the table [FIN].[dbo].[cfn\_allocation\_ho]. The results grid displays 17 rows of data with columns: Ram, CC, Name, Format, Year, Period, Period Activity, and YTD.

	Ram	CC	Name	Format	Year	Period	Period Activity	YTD
139	109	02009	02009 TX- S Ladprao120	Express	2020	07	-33892.9727080104	-213696.480331364
140	109	02009	02009 TX- S Ladprao120	Express	2020	08	-31966.7604490959	-245663.24078046
141	109	02009	02009 TX- S Ladprao120	Express	2020	09	-31452.6243920234	-277115.865172484
142	109	02009	02009 TX- S Ladprao120	Express	2020	10	-22750.951900862	-299866.817073346
143	109	02009	02009 TX- S Ladprao120	Express	2020	11	-30314.1392977524	-330180.956371098
144	109	02009	02009 TX- S Ladprao120	Express	2020	12	-29267.9091617057	-359448.865532804
145	11	01011	01011_H_Chiangmai	Hyper	2020	01	-1134295.94825485	-1134295.94825485
146	11	01011	01011_H_Chiangmai	Hyper	2020	02	-1156446.03011599	-2290741.97837084
147	11	01011	01011_H_Chiangmai	Hyper	2020	03	-1114892.4791265	-3405634.45749733
148	11	01011	01011_H_Chiangmai	Hyper	2020	04	-1123617.831534	-4529252.28903133
149	11	01011	01011_H_Chiangmai	Hyper	2020	05	-1059691.57320732	-5588943.86223865
150	11	01011	01011_H_Chiangmai	Hyper	2020	06	-1058149.62699911	-6647093.48923776

ภาพที่ 3.14 ภาพแสดงข้อมูล HO data ในโปรแกรม SSMS

5) ออกแบบตารางที่ใช้ในการเก็บข้อมูลงบกำไรขาดทุน (Profit and loss data) โดยโครงสร้างของตารางเป็นดังนี้

- คอลัมน์ที่ 1 คือ คอลัมน์ Group ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 2 คือ คอลัมน์ Acc ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 3 คือ คอลัมน์ CC ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ

- คอลัมน์ที่ 4 คือ คอลัมน์ CCT ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามแก้ไขเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- คอลัมน์ที่ 5 คือ คอลัมน์ PRD ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 6 คือ คอลัมน์ PRJ ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 7 คือ คอลัมน์ Local ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 8 คือ คอลัมน์ Inter ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 9 คือ คอลัมน์ Year ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 10 คือ คอลัมน์ Period ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 11 คือ คอลัมน์ Period Activity ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 12 คือ คอลัมน์ Period Activity YTD ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 13 คือ คอลัมน์ Period Activity LY ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 14 คือ คอลัมน์ Period Activity LY YTD ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 15 คือ คอลัมน์ Period Activity TG ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 16 คือ คอลัมน์ Period Activity TG YTD ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 17 คือ คอลัมน์ Format Original ประเภทของข้อมูลกำหนดให้เป็น varchar 50 ตัวอักษร สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 18 คือ คอลัมน์ Format ประเภทของข้อมูลกำหนดให้เป็น varchar 50 ตัวอักษร สามารถเก็บข้อมูลว่างได้

คอลัมน์ที่กำหนดให้เป็นคีย์ของตารางได้แก่คอลัมน์ Group, Acc, CC, CCT, PRD, PRJ, Local, Inter, Year, Period  
 เอกสารนี้เป็นเอกสารทสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Column Name	Data Type	Allow Nulls
[Group]	varchar(10)	<input type="checkbox"/>
Acc	varchar(10)	<input type="checkbox"/>
CC	varchar(10)	<input type="checkbox"/>
CCT	varchar(10)	<input type="checkbox"/>
PRD	varchar(10)	<input type="checkbox"/>
PRJ	varchar(10)	<input type="checkbox"/>
Local	varchar(10)	<input type="checkbox"/>
Inter	varchar(10)	<input type="checkbox"/>
Year	varchar(5)	<input type="checkbox"/>
Period	varchar(5)	<input type="checkbox"/>
Period_Activity	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_YTD	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_LY	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_LY_YTD	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_TG	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_TG_YTD	numeric(38, 2)	<input checked="" type="checkbox"/>
FormatOriginal	varchar(50)	<input checked="" type="checkbox"/>
Format	varchar(50)	<input checked="" type="checkbox"/>

ภาพที่ 3.15 ภาพแสดงโครงสร้างตาราง P&L snapshot

```

1 SELECT [Group]
2 , [Acc]
3 , [CC]
4 , [CCT]
5 , [PRD]
6 , [PRJ]
7 , [Local]
8 , [Inter]
9 , [Year]
10 , [Period]
11 , [Period_Activity]
12 , [Period_Activity_YTD]
13 , [Period_Activity_LY]
14 , [Period_Activity_LY_YTD]
15 , [Period_Activity_TG]
16 , [Period_Activity_TG_YTD]
17 , [FormatOriginal]
18 , [Format]
19 FROM [FIN].[dbo].[fpa_channel_p1_snapshot]

```

Group	Acc	CC	CCT	PRD	PRJ	Local	Inter	Year	Period	Period_Activity	Period_Activity_YTD	Period_Activity_LY	Period_Activity_LY_YTD	Period_Activity_TG	Period_Activity_TG_YTD	FormatOriginal	Format
1	081	35321	02182	0100	008	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
2	081	35321	02182	0100	400	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
3	081	35321	02183	0100	001	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
4	081	35321	02183	0100	002	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
5	081	35321	02183	0100	003	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
6	081	35321	02183	0100	004	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
7	081	35321	02183	0100	009	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
8	081	35321	02183	0100	000	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express

ภาพที่ 3.16 ภาพแสดงโครงสร้างตาราง P&L allocated snapshot

6) ออกแบบตารางที่ใช้ในการเก็บข้อมูลงบกำไรขาดทุนที่มีการกระจายรายได้และค่าใช้จ่ายส่วนกลางไปยังร้านค้าปลีก (Profit and loss allocated data) โดยโครงสร้างของตารางเป็นดังนี้

- คอลัมน์ที่ 1 คือ คอลัมน์ Group ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 2 คือ คอลัมน์ Acc ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 3 คือ คอลัมน์ CC ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 4 คือ คอลัมน์ CCT ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ

เอกสารนี้เป็นเอกสารที่เผยแพร่ให้ใช้สำหรับอาจารย์ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คอลัมน์ที่ 5 คือ คอลัมน์ PRD ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 6 คือ คอลัมน์ PRJ ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 7 คือ คอลัมน์ Local ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 8 คือ คอลัมน์ Inter ประเภทของข้อมูลกำหนดให้เป็น varchar 10 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 9 คือ คอลัมน์ Year ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 10 คือ คอลัมน์ Period ประเภทของข้อมูลกำหนดให้เป็น varchar 5 ตัวอักษร ไม่สามารถปล่อยให้ข้อมูลว่างได้ ต้องมีข้อมูลเสมอ
- คอลัมน์ที่ 11 คือ คอลัมน์ Period Activity ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 12 คือ คอลัมน์ Period Activity YTD ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 13 คือ คอลัมน์ Period Activity LY ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 14 คือ คอลัมน์ Period Activity LY YTD ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 15 คือ คอลัมน์ Period Activity TG ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 16 คือ คอลัมน์ Period Activity TG YTD ประเภทของข้อมูลกำหนดให้เป็น numeric 38 ตำแหน่ง เป็นทศนิยม 2 ตำแหน่ง สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 17 คือ คอลัมน์ Format Original ประเภทของข้อมูลกำหนดให้เป็น varchar 50 ตัวอักษร สามารถเก็บข้อมูลว่างได้
- คอลัมน์ที่ 18 คือ คอลัมน์ Format ประเภทของข้อมูลกำหนดให้เป็น varchar 50 ตัวอักษร สามารถเก็บข้อมูลว่างได้

คอลัมน์ที่กำหนดให้เป็นคีย์ของตารางได้แก่คอลัมน์ Group, Acc, CC, CCT, PRD, PRJ, Local, Inter, Year, Period  
 เอกสารนี้เป็นเอกสารทสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Column Name	Data Type	Allow Nulls
[Group]	varchar(10)	<input type="checkbox"/>
Acc	varchar(10)	<input type="checkbox"/>
CC	varchar(10)	<input type="checkbox"/>
CCT	varchar(10)	<input type="checkbox"/>
PRD	varchar(10)	<input type="checkbox"/>
PRJ	varchar(10)	<input type="checkbox"/>
Local	varchar(10)	<input type="checkbox"/>
Inter	varchar(10)	<input type="checkbox"/>
Year	varchar(5)	<input type="checkbox"/>
Period	varchar(5)	<input type="checkbox"/>
Period_Activity	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_YTD	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_LY	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_LY_YTD	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_TG	numeric(38, 2)	<input checked="" type="checkbox"/>
Period_Activity_TG_YTD	numeric(38, 2)	<input checked="" type="checkbox"/>
FormatOriginal	varchar(50)	<input checked="" type="checkbox"/>
Format	varchar(50)	<input checked="" type="checkbox"/>

ภาพที่ 3.17 ภาพแสดงโครงสร้างตาราง P&L allocated snapshot

```

1 | SELECT [Group]
2 | , [Acc]
3 | , [CC]
4 | , [CCT]
5 | , [PRD]
6 | , [PRJ]
7 | , [Local]
8 | , [Inter]
9 | , [Year]
10 | , [Period]
11 | , [Period_Activity]
12 | , [Period_Activity_YTD]
13 | , [Period_Activity_LY]
14 | , [Period_Activity_LY_YTD]
15 | , [Period_Activity_TG]
16 | , [Period_Activity_TG_YTD]
17 | , [FormatOriginal]
18 | , [Format]]
19 | FROM [FIN].[dbo].[p_l_allocate_snapshot]
20 |

```

Group	Acc	CC	CCT	PRD	PRJ	Local	Inter	Year	Period	Period_Activity	Period_Activity_YTD	Period_Activity_LY	Period_Activity_LY_YTD	Period_Activity_TG	Period_Activity_TG_YTD	FormatOriginal	Format
980	081	35321	02377	0100	400	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
981	081	35321	02378	0100	001	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
982	081	35321	02378	0100	002	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
983	081	35321	02378	0100	003	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
984	081	35321	02378	0100	004	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express
985	081	35321	02378	0100	009	000000	00000	000	2021	12	0.00	0.00	0.00	0.00	0.00	Express	Express

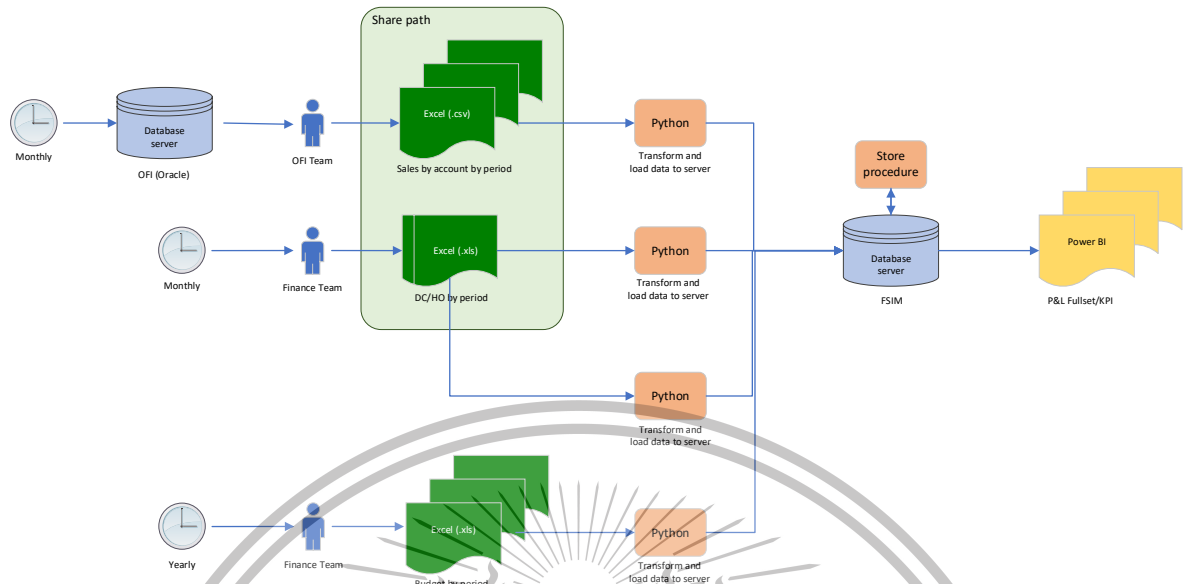
ภาพที่ 3.18 ภาพแสดงโครงสร้างตาราง P&L allocated snapshot

### 3.4.2 ออกแบบและพัฒนาระบบการไหลของข้อมูล

ออกแบบการไหลของข้อมูลโดยเริ่มจากการรวบรวมข้อมูลดิบจากฝ่ายต่าง ๆ ที่เกี่ยวข้อง ข้อมูลที่ได้มาเป็นข้อมูลดิบที่ยังไม่ถูกปรับแต่งให้อยู่ในรูปแบบที่สามารถนำไปใช้งานต่อได้ ดังนั้น ก่อนที่จะไหลของข้อมูลเหล่านี้เก็บในเซิร์ฟเวอร์ฐานข้อมูลจึงจะใช้โปรแกรมที่ชื่อว่า Jupyter Notebook ทำการปรับเปลี่ยนและทำความสะอาดข้อมูลให้อยู่ในรูปแบบที่พร้อมเก็บลงในเซิร์ฟเวอร์ฐานข้อมูล ซึ่งภายในเซิร์ฟเวอร์ฐานข้อมูลข้อมูลจะถูกปรับเปลี่ยนอีกครั้งเพื่อให้ได้ข้อมูลที่มีความหมายมากยิ่งขึ้น จากนั้นจะใช้โปรแกรม Microsoft Power BI นำข้อมูลที่ได้ไปสร้างรายงานงบกำไรขาดทุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

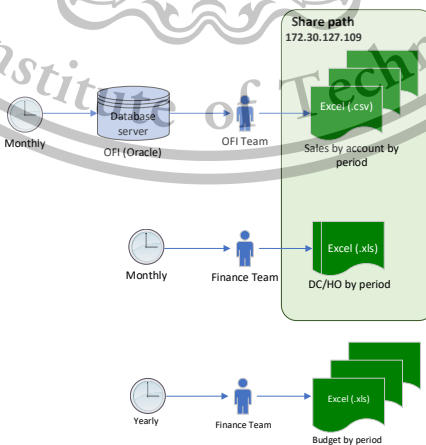




ภาพที่ 3.19 ภาพแสดงการไหลของข้อมูลในการจัดทำรายงานงบกำไรขาดทุน

1) รับข้อมูลดิบจากฝ่ายอื่น

ข้อมูลที่เป็นต่อการจัดทำรายงานงบกำไรขาดทุนที่ได้มาจากฝ่ายอื่นในบริษัทได้แก่ ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริง (Actual data) ข้อมูลรายได้และค่าใช้จ่ายที่คาดว่าจะเกิดขึ้น (Budget data) ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นที่ศูนย์กระจายสินค้า (DC data) และข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นที่สำนักงานใหญ่ (HO data) ข้อมูลเหล่านี้ได้มาจากฝ่าย OFI และฝ่าย FPA-channel ซึ่งทั้งสองทีมจะมีการนำไฟล์ข้อมูลเหล่านี้ ยกเว้นข้อมูลรายได้และค่าใช้จ่ายที่คาดว่าจะเกิดขึ้น รวบรวมวางไว้ที่โฟลเดอร์ส่วนกลางเป็นประจำทุกเดือน ส่วนข้อมูลรายได้และค่าใช้จ่ายที่คาดว่าจะเกิดขึ้นนั้นจะถูกส่งให้ทางอีเมลปีละครั้งเท่านั้น



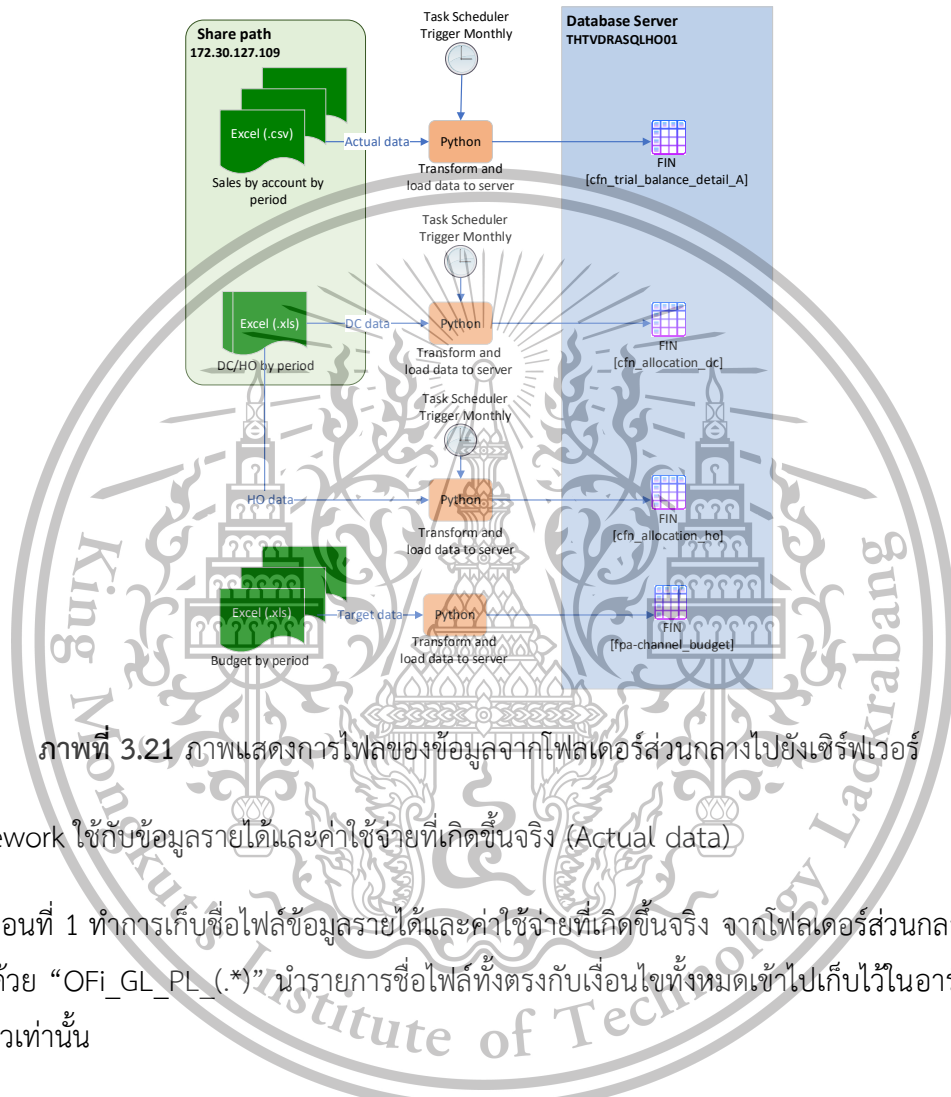
ภาพที่ 3.20 ภาพแสดงวิธีการรับข้อมูลดิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น การนำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 2) ทำความสะอาด แปลง และโหลดข้อมูลเข้าไปยังเซิร์ฟเวอร์ฐานข้อมูล

การทำความสะอาด แปลง และโหลดข้อมูลจะกระทำโดยใช้โปรแกรม Jupyter Notebook ทั้งหมด โดยมี ไฟล์ .py ที่ใช้ 4 ไฟล์ด้วยกัน



ภาพที่ 3.21 ภาพแสดงการไหลของข้อมูลจากไฟล์เครื่องส่วนกลางไปยังเซิร์ฟเวอร์

- ไฟล์ framework ใช้กับข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริง (Actual data)

ขั้นตอนที่ 1 ทำการเก็บชื่อไฟล์ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริง จากไฟล์เครื่องส่วนกลาง คืออ่านชื่อไฟล์ที่ขึ้นต้นด้วย "OFi\_GL\_PL\_(\*)." นำรายการชื่อไฟล์ทั้งหมดตรงกับเงื่อนไขทั้งหมดเข้าไปเก็บไว้ในอาร์เรย์ ซึ่งควรมีเพียงไฟล์เดียวเท่านั้น

```
path = r'\\172.30.127.109\Reporting\PL'
files = os.listdir(path)
files_list = []
prep_files = []
for f in files:
    result_list = re.findall(r'OFi_GL_PL_(*).', f)
    files_list.append(result_list)
```

ภาพที่ 3.22 ภาพแสดงโค้ด framework (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 ทำการอ่านไฟล์ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริง และโหลดข้อมูลเข้า dataframe

```
source = files_list[0]
df = pd.read_csv(source, delimiter="|", header=None, dtype = str)
query_job_content_valid(df)
```

ภาพที่ 3.23 ภาพแสดงโค้ด framework (2)

ขั้นตอนที่ 3 ทำการแปลงข้อมูล ได้แก่ เพิ่มชื่อให้คอลัมน์ ระบุเดือนและปีจากเนื้อหาของข้อมูล แยกเป็นคอลัมน์เดือนกับคอลัมน์ปี เรียงลำดับคอลัมน์ใหม่

```
df.columns = ["Period-Year", "Group", "Acc", "CC", "CCT", "PRD", "PRJ", 'Local',
'Inter', 'Beginning Balance', 'Period Activity', 'Ending Balance', 'Currency']
df['Period'] = df['Period-Year'].str.slice(0,2)
df['Year'] = '20' + df['Period-Year'].str.slice(3,5)
df =
df[['Group', 'Acc', 'CC', 'CCT', 'PRD', 'PRJ', 'Local', 'Inter', 'Year', 'Period', 'Beginning
Balance', 'Period Activity', 'Ending Balance']]
```

ภาพที่ 3.24 ภาพแสดงโค้ด framework (3)

ขั้นตอนที่ 4 คำนวณข้อมูล

```
df[["Beginning Balance", "Ending Balance", "Period Activity"]] = df[["Beginning
Balance", "Ending Balance", "Period Activity"]].astype(float)
total1 = int(df["Beginning Balance"].sum())
total2 = int(df["Period Activity"].sum())
total3 = int(df["Ending Balance"].sum())
```

ภาพที่ 3.25 ภาพแสดงโค้ด framework (4)

ขั้นตอนที่ 5 เชื่อมต่อกับเซิร์ฟเวอร์ฐานข้อมูลที่ต้องการ

```
server = 'THTVDRASQLH001'
database = 'FIN'
conn_string = 'mssql+pyodbc://{}/{}?driver=SQL Server?
Trusted_Connection=yes'.format(server, database)
engine = sa.create_engine(conn_string, encoding="utf8")
```

ภาพที่ 3.26 ภาพแสดงโค้ด framework (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

ขั้นตอนที่ 6 นำข้อมูลที่ถูกแปลงเรียบร้อยแล้วโหลดเข้าเซิร์ฟเวอร์ฐานข้อมูล โดยมีการกำหนดรูปแบบของข้อมูลด้วย

```
df.to_sql(DEST_TABLE,
engine,if_exists=method,index=False,schema="dbo",chunksize = 5000,
dtype = {'Group':sal.types.VARCHAR(length=50),
'Acc':sal.types.VARCHAR(length=50),
'CC':sal.types.VARCHAR(length=50),
'CCT':sal.types.VARCHAR(length=50),
'PRD':sal.types.VARCHAR(length=50),
'PRJ':sal.types.VARCHAR(length=50),
'Local':sal.types.VARCHAR(length=50),
'Inter':sal.types.VARCHAR(length=50),
'Year':sal.types.VARCHAR(length=50),
'Period':sal.types.VARCHAR(length=50),
'Beginning Balance':sal.types.Float,
'Period Activity':sal.types.Float,
'Ending Balance':sal.types.Float})
```

ภาพที่ 3.27 ภาพแสดงโค้ด framework (6)

- ไฟล์ framework DC ใช้กับข้อมูลรายได้และค่าใช้จ่ายที่เกิดที่ศูนย์กระจายสินค้า (DC data)

ขั้นตอนที่ 1 ทำการเก็บชื่อไฟล์ข้อมูลรายได้และค่าใช้จ่ายที่เกิดที่ศูนย์กระจายสินค้า จากโพลเดอร์ ส่วนกลาง คืออ่านชื่อไฟล์ที่ขึ้นต้นด้วย “Net DC & HO Y2021\_Send\_” นำรายการชื่อไฟล์ทั้งตรงกับเงื่อนไขทั้งหมดเข้าไปเก็บไว้ในอาร์เรย์ ซึ่งควรมีเพียงไฟล์เดียวเท่านั้น

```
rawString = r'\\172.30.127.109\Reporting\PL'
fileName = [f for f in listdir(rawString) if isfile(join(rawString, f))] # Getting
useFile = []
# Getting the file we want from path
for i in range(len(fileName)):
    if fileName[i][0:23] == "Net DC & HO Y2021_Send_":
        useFile.append(fileName[i])
    else:
        pass
```

ภาพที่ 3.28 ภาพแสดงโค้ด framework DC (1)

ขั้นตอนที่ 2 ทำการอ่านไฟล์ข้อมูลรายได้และค่าใช้จ่ายที่เกิดที่ศูนย์กระจายสินค้า ไฟล์ที่ได้มาจะมีข้อมูลของทั้งข้อมูลรายได้และค่าใช้จ่ายที่เกิดที่ศูนย์กระจายและที่สำนักงานใหญ่ จึงทำการเก็บชื่อชิตในไฟล์แล้วเรียงลำดับชื่อชิตนั้น โดยที่ชิตข้อมูลของศูนย์กระจายสินค้าจะถูกเก็บอยู่ในตำแหน่งที่ 1 และโหลดข้อมูลเข้า dataframe

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

```

# Excel file to dataframe
xls = pd.ExcelFile(rawString + "\\\" + useFile[0])
# Get sheet name and sort sheet name according to correct order H0 then DC
sheetNames = xls.sheet_names
sheetNames = sorted(sheetNames, reverse= True)
dataframeList = []
for i in sheetNames:
    df = pd.read_excel(xls, i, index_col = False, skiprows =[0,1,2,3,4,5,6,7,8,9],
usecols=[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15],dtype='str')
    dataframeList.append(df)

```

ภาพที่ 3.29 ภาพแสดงโค้ด framework DC (2)

ขั้นตอนที่ 3 ทำการแปลงข้อมูล ได้แก่ เปลี่ยนชื่อให้คอลัมน์ กรองข้อมูลที่ไม่ต้องการออก ระบุปีจากชื่อไฟล์ แยก เรียงลำดับคอลัมน์ใหม่ ทำการเปลี่ยนแกนของข้อมูลโดยให้ข้อมูลอยู่ในคอลัมน์ Period

```

df2 = pd.DataFrame(create_key()[3][1])
#-- Rename column --#
df2.rename(columns={'P1':'01', 'P2':'02', 'P3':'03', 'P4':'04', 'P5':'05', 'P6':'06',
'P7':'07', 'P8':'08', 'P9':'09', 'P10':'10', 'P11':'11', 'P12':'12'}, inplace=True)
#-- filter rows --#
df = df2[df2['RMS'].notnull()]
df = df[df['RMS'] != 'nan']
query_job_content_valid(df)
print("Expected Number of rows: " + str(len(df)*12))

#-- get year --#
df["Year"] = str(useFile[13:16])
#-- unpivot dataframe --#
df = df.melt(['RMS', 'CC', 'Name', 'Format', 'Year'], var_name = 'Period',
value_name='Period Activity')

```

ภาพที่ 3.30 ภาพแสดงโค้ด framework DC (3)

ขั้นตอนที่ 4 เปลี่ยนค่าที่เป็นค่าว่างให้เป็น 0 กำหนดรูปแบบของข้อมูล และสร้างคอลัมน์ใหม่

```

#-- change column type --#
df['Period Activity'] = df['Period Activity'].replace(['nan'], '0')
df['Period Activity'] = df['Period Activity'].astype(float)
# Create YTD column --#
df['YTD'] = round(0.0,11)

```

ภาพที่ 3.31 ภาพแสดงโค้ด framework DC (4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ขั้นตอนที่ 5 คำนวณค่าข้อมูลให้เป็นค่าข้อมูลแบบ year to date โดยทำการ

```
#-- calculate YTD --#  
  
for index, row in df.iterrows():  
    period = int(row['Period'])  
    period_activity = float(row['Period Activity'])  
  
    #reset sum after > P 12  
    if period == 1:  
        sum_ytd = period_activity  
    else:  
        sum_ytd = sum_ytd + period_activity  
  
df.at[index, 'YTD'] = sum_ytd
```

ภาพที่ 3.32 ภาพแสดงโค้ด framework DC (5)

ขั้นตอนที่ 6 เชื่อมต่อกับเซิร์ฟเวอร์ฐานข้อมูลที่ต้องการ

```
server = 'THTVDRASQLH001'  
database = 'FIN'  
conn_string = 'mssql+pyodbc://{}/{}?driver=SQL Server?  
Trusted_Connection=yes'.format(server, database)  
engine = sa.create_engine(conn_string, encoding="utf8")
```

ภาพที่ 3.33 ภาพแสดงโค้ด framework DC (6)

ขั้นตอนที่ 7 นำข้อมูลที่ถูกแปลงเรียบร้อยแล้วโหลดเข้าเซิร์ฟเวอร์ฐานข้อมูล

```
df.to_sql(DEST_TABLE, engine, if_exists='append', index=False, schema="dbo")
```

ภาพที่ 3.34 ภาพแสดงโค้ด framework DC (7)

- ไฟล์ framework HO ใช้กับข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นที่สำนักงานใหญ่ (HO data)

ขั้นตอนที่ 1 ทำการเก็บชื่อไฟล์ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นที่ศูนย์กระจายสินค้า จากโฟลเดอร์ ส่วนกลาง คืออ่านชื่อไฟล์ที่ขึ้นต้นด้วย “Net DC & HO Y2021\_Send\_” นำรายการชื่อไฟล์ทั้งหมดตรงกับเงื่อนไข ทั้งหมดเข้าไปเก็บไว้ในอาร์เรย์ ซึ่งควรมีเพียงไฟล์เดียวเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

rawString = r'\\172.30.127.109\Reporting\PL'
fileName = [f for f in listdir(rawString) if.isfile(join(rawString, f))] # Getting
useFile = []
# Getting the file we want from path
for i in range(len(fileName)):
    if fileName[i][0:23] == "Net DC & HO Y2021_Send_":
        useFile.append(fileName[i])
    else:
        pass

```

ภาพที่ 3.35 ภาพแสดงโค้ด framework HO (1)

ขั้นตอนที่ 2 ทำการอ่านไฟล์ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นที่ศูนย์กระจายสินค้า ไฟล์ที่ได้มาจะมีข้อมูลของทั้งข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นที่ศูนย์กระจายและที่สำนักงานใหญ่ จึงทำการเก็บชื่อชีทในไฟล์แล้วเรียงลำดับชื่อชีทนั้น โดยที่ชื่อข้อมูลของสำนักงานใหญ่จะถูกเก็บอยู่ในตำแหน่งที่ 0 และโหลดข้อมูลเข้า

```

# Excel file to dataframe
xls = pd.ExcelFile(rawString + "\\*"+ useFile[0])
# Get sheet name and sort sheet name according to correct order HO then DC
sheetNames = xls.sheet_names
sheetNames = sorted(sheetNames, reverse=True)
dataframeList = []
for i in sheetNames:
    df = pd.read_excel(xls, i, index_col = False, skiprows =[0,1,2,3,4,5,6,7,8,9],
    usecols=[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15], dtype='str')
    dataframeList.append(df)

```

ภาพที่ 3.36 ภาพแสดงโค้ด framework HO (2)

ขั้นตอนที่ 3 ทำการแปลงข้อมูล ได้แก่ เปลี่ยนชื่อให้คอลัมน์ กรองข้อมูลที่ไม่ต้องการออก ระบุปีจากชื่อไฟล์ แยก เรียงลำดับคอลัมน์ใหม่ ทำการเปลี่ยนแกนของข้อมูลโดยให้ข้อมูลอยู่ในคอลัมน์ Period

```

df2 = pd.DataFrame(create_key()[3][0])
#-- Rename column --#
df2.rename(columns={'P1':'01', 'P2':'02', 'P3':'03', 'P4':'04', 'P5':'05', 'P6':'06',
'P7':'07', 'P8':'08', 'P9':'09', 'P10':'10', 'P11':'11', 'P12':'12'}, inplace=True)
#-- filter rows --#
df = df2[df2['Ram'].notnull()]
df = df[df['Ram'] != 'nan']
query_job_content_valid(df)
print("Expected Number of rows: " + str(len(df)*12))

#-- get year --#
df["Year"] = str(useFile[13:16])
#-- unpivot dataframe --#
df = df.melt(['Ram', 'CC', 'Name', 'Format', 'Year'], var_name = 'Period',
value_name='Period Activity')

```

ภาพที่ 3.37 ภาพแสดงโค้ด framework HO (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูผู้ใช้งานที่ออกเสียงมาเท่านั้น โปรดอย่าให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 เปลี่ยนค่าที่เป็นค่าว่างให้เป็น 0 กำหนดรูปแบบของข้อมูล และสร้างคอลัมน์ใหม่

```
#-- change column type --#
df['Period Activity'] = df['Period Activity'].replace(['nan'], '0')
df['Period Activity'] = df['Period Activity'].astype(float)
# Create YTD column --#
df['YTD'] = round(0.0,11)
```

ภาพที่ 3.38 ภาพแสดงโค้ด framework HO (4)

ขั้นตอนที่ 5 คำนวณค่าข้อมูลให้เป็นค่าข้อมูลแบบ year to date โดยทำการ

```
#-- calculate YTD --#
for index, row in df.iterrows():
    period = int(row['Period'])
    period_activity = float(row['Period Activity'])

    #reset sum after > P 12
    if period == 1:
        sum_ytd = period_activity
    else:
        sum_ytd = sum_ytd + period_activity

    df.at[index, 'YTD'] = sum_ytd
```

ภาพที่ 3.39 ภาพแสดงโค้ด framework HO (5)

ขั้นตอนที่ 6 เชื่อมต่อกับเซิร์ฟเวอร์ฐานข้อมูลที่ต้องการ

```
server = 'THTVDRASQLH001'
database = 'FIN'
conn_string = 'mssql+pyodbc://{}/{}?driver=SQL Server?
Trusted_Connection=yes'.format(server, database)
engine = sa.create_engine(conn_string, encoding="utf8")
```

ภาพที่ 3.40 ภาพแสดงโค้ด framework HO (6)

ขั้นตอนที่ 7 นำข้อมูลที่ถูกลบเรียบร้อยแล้วโหลดเข้าเซิร์ฟเวอร์ฐานข้อมูล

```
df.to_sql(DEST_TABLE, engine, if_exists='append', index=False, schema="dbo")
```

ภาพที่ 3.41 ภาพแสดงโค้ด framework HO (7)

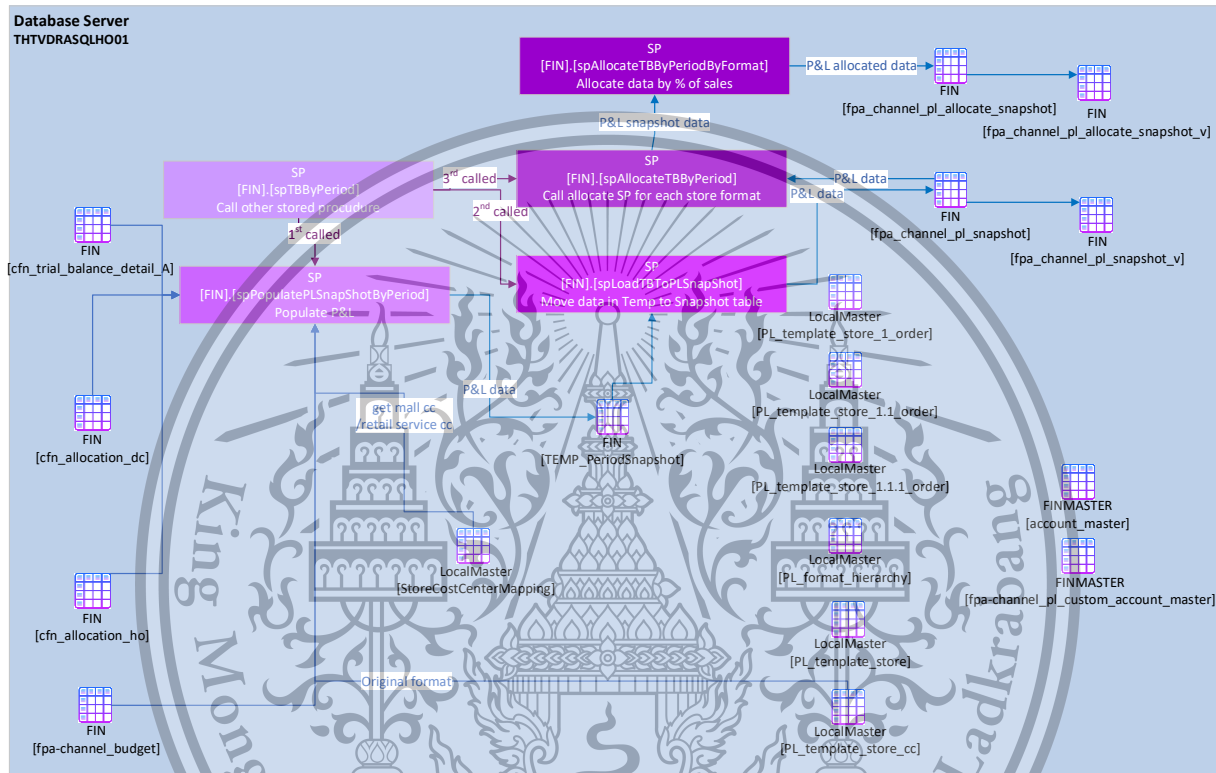
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

### 3) แปลงข้อมูลที่ทำให้ความสะอาดเรียบร้อยแล้วในโปรแกรม SSMS

หลังจากที่ข้อมูลถูกโหลดเข้ามาในฐานข้อมูลแล้ว ข้อมูลจะถูกแปลงต่อด้วย stored procedure ที่เขียนขึ้น มีการนำเอาตารางข้อมูลอื่น ๆ มาใช้ร่วมด้วย เพื่อทำการกรองข้อมูล คำนวณข้อมูล เชื่อมโยงข้อมูลรายได้ ค่าใช้จ่ายกับข้อมูลอื่นให้ได้ข้อมูลมากยิ่งขึ้น



ภาพที่ 3.42 ภาพแสดงการไหลของข้อมูลภายในเซิร์ฟเวอร์ฐานข้อมูล

นอกจากนี้ภายใน stored procedure ยังมีการใช้ฟังก์ชันอีกด้วย มี 2 ฟังก์ชัน ดังนี้

- GetAccountMappingForPL เป็นคอลัมน์ที่ใช้ในการเปลี่ยนข้อมูลของคอลัมน์ Acc โดยมีการดูเงื่อนไขจากคอลัมน์อื่น ๆ ในตาราง เพื่อให้ได้มุมมองของกลุ่มของรายได้และค่าใช้จ่ายครบถ้วนตามที่คุณใช้งานต้องการ

ฟังก์ชันนี้มีการรับพารามิเตอร์เข้ามา 4 ตัว ได้แก่ CCT, Acc, PRD และ Local จากนั้นสร้างตัวแปรชื่อ custom\_account ขึ้นมา เอาไว้เก็บข้อมูล Acc ที่ได้ค่าขึ้นมาใหม่จากเงื่อนไขที่เขียนขึ้นในฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





- GetFormatMappingForPL เป็นคอลัมน์ที่ใช้ในการเปลี่ยนข้อมูลของคอลัมน์ Format โดยมีการดูเงื่อนไขจากคอลัมน์อื่น ๆ ในตาราง เพื่อให้ได้มุมมองของประเภทของร้านค้าปลีกตามที่ผู้ใช้งานต้องการ

ฟังก์ชันนี้มีการรับพารามิเตอร์เข้ามา 3 ตัว ได้แก่ CCT, CC และ original\_format จากนั้นสร้างตัวแปรชื่อ custom\_format ขึ้นมา เอาไว้เก็บข้อมูล Format ที่ได้ค่าขึ้นมาใหม่จากเงื่อนไขที่เขียนขึ้นในฟังก์ชัน

```
-- CLR Scalar Function Syntax
ALTER FUNCTION [dbo].[GetFormatMappingForPL] (
    @cct varchar(4),
    @cc varchar(6),
    @original_format varchar(20)
)
RETURNS varchar(25)
AS
BEGIN

    DECLARE @custom_format AS varchar(25);
    SELECT @custom_format = CASE

        WHEN @cct = '0100' THEN 'Pre-opening'
        WHEN @cct = '0600' THEN 'Cost of store closure'
        WHEN @cct = '0800' THEN 'DC'
        WHEN @cct = '0900' AND @cc = '09899' THEN 'HO-GHS'
        WHEN @cct = '0900' AND @cc = '09240' THEN 'HO-WKP'
        WHEN @cct = '0900' AND @cc = '09660' THEN 'HO-B2B'
        WHEN @cct = '0900' AND @cc = '09220' THEN 'HO-NC'
        WHEN @cct = '0900' THEN 'HO'

        WHEN (@original_format IS NULL) AND (@cct = '0100') THEN
            CASE
                WHEN CAST(@cc AS numeric(18)) IN (31020, 31016, 31017, 31500, 30015) THEN 'GHS-Extra'
                WHEN CAST(@cc AS numeric(18)) IN (31000, 31021, 31038, 31009, 31011, 31033, 30010) THEN 'GHS-Hyper'
                ELSE 'GHS-Others'
            END
    END
```

ภาพที่ 3.45 ภาพแสดงโค้ดของฟังก์ชัน GetFormatMappingForPL (1)

เมื่อทำการหาค่าเสร็จแล้ว ค่า Format ใหม่ที่ได้ถูกเก็บอยู่ในตัวแปร custom\_format และตัวแปรนี้จะถูกส่งค่ากลับไป

```
WHEN (@cct = '0100') THEN -- Online: Marketplace
CASE
    WHEN CAST(@cc AS numeric(18)) IN (5902, 5903) THEN 'HappyFresh'
    WHEN CAST(@cc AS numeric(18)) = 5904 THEN 'Shopee'
    WHEN CAST(@cc AS numeric(18)) IN (5900, 5901, 5903) THEN 'Laazda'
    WHEN CAST(@cc AS numeric(18)) = 5907 THEN 'FoodPanda'
    WHEN CAST(@cc AS numeric(18)) = 5909 THEN 'WeFresh'
    WHEN CAST(@cc AS numeric(18)) IN (5905, 5906) THEN 'WKP-Others'

    WHEN CAST(@cc AS numeric(18)) = 1000 THEN 'Hyper-Pool'
    WHEN CAST(@cc AS numeric(18)) = 2000 THEN 'Express-Pool'
    WHEN CAST(@cc AS numeric(18)) = 1500 THEN 'Extra-Pool'
    WHEN CAST(@cc AS numeric(18)) = 4000 THEN 'Supermarket-Pool'
    WHEN CAST(@cc AS numeric(18)) = 5500 THEN 'Department Store-Pool'
    WHEN CAST(@cc AS numeric(18)) = 5700 THEN 'B2B-Pool'

    WHEN CAST(@cc AS numeric(18)) IN (1, 7, 9, 10, 12, 56, 57, 58, 59, 60, 63, 64, 65, 90, 1129, 1143, 1144) THEN 'Hypermarket'
    WHEN CAST(@cc AS numeric(18)) IN (2, 16, 20, 61) THEN 'Express'
    WHEN CAST(@cc AS numeric(18)) IN (11, 15) THEN 'Extra'
    WHEN CAST(@cc AS numeric(18)) IN (13, 55, 5531, 5540, 5541) THEN 'Department Store'
    WHEN CAST(@cc AS numeric(18)) IN (14, 40, 4034, 4173) THEN 'Supermarket'
    WHEN CAST(@cc AS numeric(18)) IN (5954) THEN 'B2B'
    ELSE @original_format
END

ELSE @original_format

END

--set @custom_account = @CCT
RETURN @custom_format;
END
```

ภาพที่ 3.46 ภาพแสดงโค้ดของฟังก์ชัน GetFormatMappingForPL (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย Stored procedure มีรายละเอียดดังนี้

- spTBByPeriod เป็น Stored procedure ที่สร้างขึ้นเพื่อใช้เรียก Stored procedure อื่นเรียงตามลำดับ

Stored procedure นี้มีการรับพารามิเตอร์ปีและเดือน แต่เป็นพารามิเตอร์ที่ไม่จำเป็นต้องส่งค่าเข้ามาก็ได้ หากไม่มีค่าส่งเข้ามาปีหรือเดือนนั้นจะถูกกำหนดให้เป็นค่าว่าง เมื่อปีหรือเดือนเป็นค่าว่างโปรแกรมจะทำการหาค่าจากวันที่ ณ ตอนที่รัน Stored procedure

```
-- =====
-- Author: <Haruehai A.>
-- Create date: <Sep 21, 2020>
-- Description: <This store procedure contain other store procedures, prepare P&L data for PowerBI>
-- Parameters: @year, @period

-- EXEC [FIN].[dbo].[spTBByPeriod] @year='2021', @period='06'

-- =====
ALTER PROCEDURE [dbo].[spTBByPeriod] (@year as varchar(4) = NULL, @period as varchar(2) = NULL)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    IF @year IS NULL
    BEGIN
        SET @year = ( select year([CalendarDate]) [Year]
                    from [RMSMASTER].[dbo].[Calendar]
                    where YearNo = year(GETDATE()) and month([CalendarDate]) = month(GETDATE()) and day([CalendarDate]) = day(GETDATE()))
    END

    IF @period IS NULL
    BEGIN
        SET @period = ( select [Period]
                      from [RMSMASTER].[dbo].[Calendar]
                      where YearNo = year(GETDATE()) and month([CalendarDate]) = month(GETDATE()) and day([CalendarDate]) = day(GETDATE()))
    END
END
```

ภาพที่ 3.47 ภาพแสดงโค้ดของ spTBByPeriod (1)

เมื่อได้ปีและเดือนมาแล้ว โปรแกรมจะทำการลบข้อมูลที่ตารางที่มีการเก็บข้อมูล P&L snapshot (fpa\_channel\_pl\_snapshot) ซึ่งลบเฉพาะข้อมูลของปีและเดือนที่มีการรับค่ามาเท่านั้น จากนั้นจึงสั่งรัน Stored procedure อีก 3 ตัว โดยจะส่งพารามิเตอร์ปีและเดือนเดียวกันไปที่ทุก Stored procedure

```
DELETE [FIN].[dbo].[fpa_channel_pl_snapshot] WHERE [Year] = @year and [Period] = @period
PRINT N'Data Year = ' + @year + ' Period = ' + @period

-- Part 1: Mapping Account/Format and prepare data
EXEC [FIN].[dbo].[spPopulatePLSnapshotByPeriod] @year = @year, @period = @period

-- Part 2: Create PL snapshot
EXEC [FIN].[dbo].[spLoadTBToPLSnapshot] @year = @year, @period = @period

-- Part 3: Create PL allocation snapshot
EXEC [FIN].[dbo].[spAllocateTBByPeriod] @year = @year, @period = @period

END
GO
```

ภาพที่ 3.48 ภาพแสดงโค้ดของ spTBByPeriod (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- spPopulatePLSnapshotByPeriod เป็น Stored procedure ที่สร้างขึ้นเพื่อจัดการข้อมูลให้เป็นข้อมูลที่อยู่ในรูปแบบของข้อมูลงบกำไรขาดทุน

Stored procedure นี้มีการรับพารามิเตอร์ปีและเดือนเข้ามา จากนั้นทำการเตรียมตารางไว้ใช้งาน โดยการลบตาราง TEMP\_PeriodSnapshot

```

-- =====
-- Author: <Haruethai A.>
-- Create date: <Sep 4, 2020>
-- Description: <This SP loads trial balance of the specified (year,period) into the P&L-ready format>

-- exec [spPopulatePLSnapshotByPeriod] @group = '081', @year = 2021, @period = 5

-- =====
ALTER PROCEDURE [dbo].[spPopulatePLSnapshotByPeriod] (@year as integer = NULL, @period as integer = NULL)
AS
BEGIN
-- SET NOCOUNT ON added to prevent extra result sets from
-- interfering with SELECT statements.
SET NOCOUNT ON;

IF OBJECT_ID('['FIN].[dbo].[TEMP_PeriodSnapshot]') IS NOT NULL DROP TABLE [dbo].[TEMP_PeriodSnapshot]

```

ภาพที่ 3.49 ภาพแสดงโค้ดของ spPopulatePLSnapshotByPeriod (1)

ขั้นตอนที่ 1 นำข้อมูลของรายได้ค่าใช้จ่ายที่เป็นส่วนจากร้านค้าปลีกที่มีข้อมูลที่ต้องการครบแล้ว ใส้ไปในตาราง TEMP\_PeriodSnapshot ที่เตรียมไว้

```

-- STEP 1: Insert normal TB, except (Mail, Retail Service, PC, HQ)
select
tb.[Group]
,[FIN].[dbo].[GetAccountMappingForPL] (tb.[CCT], tb.[ACC], tb.[PRD], tb.[Local]) AS [ACC]
,tb.[CC]
,tb.[CCT]
,tb.[PRD]
,tb.[Local]
,tb.[Inter]
,tb.[Year]
,tb.[Period]
,SUM(isnull(cast(tb.[actual] as numeric(18,2)),0) * 1) as 'Period Activity'
,SUM(isnull(cast(tb.[actual_ytd] as numeric(18,2)),0) * 1) as 'Period Activity YTD'
,SUM(isnull(cast(tb.[actual_ly] as numeric(18,2)),0) * 1) as 'Period Activity LY'
,SUM(isnull(cast(tb.[actual_ly_ytd] as numeric(18,2)),0) * 1) as 'Period Activity LY YTD'
,SUM(isnull(cast(tb.[budget] as numeric(18,2)),0) * 1) as 'Period Activity TG'
,SUM(isnull(cast(tb.[budget_ytd] as numeric(18,2)),0) * 1) as 'Period Activity TG YTD'
,s.StoreFormatName AS [FormatOriginal]
,[FIN].[dbo].[GetFormatMappingForPL] (tb.[CCT], tb.[CC], s.StoreFormatName) AS [Format]
INTO [FIN].[dbo].[TEMP_PeriodSnapshot]
FROM
(
-- this year
SELECT [Group], cc, cct, acc, prd, local, prj, inter, year, period, [Period Activity] as actual, [Ending Balance] as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE year = @year and [Period] = @period and [ACC] = '30000'

UNION ALL
-- budget
SELECT [Group], cc, cct, acc, prd, local, prj, inter, year, period, '0' as actual, '0' as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, [Period Activity] as budget, [YTD] as budget_ytd
FROM [FIN].[dbo].[fpa_channel_budget] b
WHERE year = @year and [Period] = @period and [ACC] = '30000'

UNION ALL
-- last year
SELECT [Group], cc, cct, acc, prd, local, prj, inter, @year, period, '0' as actual, '0' as actual_ytd, [Period Activity] as actual_ly, [Ending Balance] as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE year = @year - 1 and [Period] = @period and [ACC] >= '30000'
) as tb
LEFT JOIN [LocalMaster].[dbo].[PL_template_store_cc] s ON s.CC = tb.CC
WHERE tb.[CCT] != '0130' AND tb.[CCT] != '0141' AND tb.[CCT] != '0880' AND tb.[CCT] != '0900'

```

ภาพที่ 3.50 ภาพแสดงโค้ดของ spPopulatePLSnapshotByPeriod (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 นำข้อมูลของรายได้ค่าใช้จ่ายที่เป็นส่วนของศูนย์การค้าที่มีข้อมูลที่ต้องการครบแล้ว ใส้ไปในตาราง TEMP\_PeriodSnapshot

```
-- STEP 2: Insert Mall (Mall: distribute mall to proper format (1) Change 'format' from Mall to proper-retail format (2) Change 'cc' from MallCC to Retail CC)
INSERT INTO [FIN].[dbo].[TEMP_PeriodSnapshot]
SELECT [Group],
[FIN].[dbo].[GetAccountMappingForPL] (tb.[CCT], tb.[ACC], tb.[PRD], tb.[Local]) AS [ACC]
,CASE
WHEN (tb.cc = '21000') THEN '01000'
WHEN (tb.cc = '21500') THEN '01500'
WHEN (tb.cc = '24000') THEN '04000'
WHEN (tb.cc = '25500') THEN '05500'
WHEN (tb.cc in ('24501', '24502', '24503', '24504', '22000')) THEN '02000'
ELSE s.[cc]
END AS [cc]
,tb.[CCT]
,tb.[PRD]
,tb.[PRJ]
,tb.[Local]
,tb.[Inter]
,tb.[Year]
,tb.[Period]
,SUM(cast(tb.[actual] as numeric(18,2)) * -1) as 'Period Activity'
,SUM(cast(tb.[actual_ytd] as numeric(18,2)) * -1) as 'YTD'
,SUM(cast(tb.[actual_ly] as numeric(18,2)) * -1) as 'Period Activity LY'
,SUM(cast(tb.[actual_ly_ytd] as numeric(18,2)) * -1) as 'Period Activity LY YTD'
,SUM(cast(tb.[budget] as numeric(18,2)) * 1) as 'Period Activity TG'
,SUM(cast(tb.[budget_ytd] as numeric(18,2)) * 1) as 'Period Activity TG YTD'
,s.StoreFormatName AS [FormatOriginal]
,CASE
WHEN (tb.cc = '20000') THEN 'Property Agent'
WHEN (tb.cc = '21000') THEN 'Hyper-Pool'
WHEN (tb.cc = '21500') THEN 'Extra-Pool'
WHEN (tb.cc = '24000') THEN 'Supermarket-Pool'
WHEN (tb.cc = '25500') THEN 'Department Store-Pool'
WHEN (tb.cc in ('24501', '24502', '24503', '24504', '22000')) THEN 'Express-Pool'
ELSE [FIN].[dbo].[GetFormatMappingForPL] (tb.[CCT])u s.[cc],s.StoreFormatName)
END AS [Format]
FROM (
-- this year
SELECT [Group],cc, cct, acc, prd, local, prj, Inter, year, period, [Period Activity] as actual, [Ending Balance] as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year and [Period] = @period and [ACC] >= '30000' and cct = '0130'

UNION ALL
-- budget
SELECT [Group],cc, cct, acc, prd, local, prj, Inter, year, period, '0' as actual, '0' as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, [Period Activity] as budget, [YTD] as budget_ytd
FROM [FIN].[dbo].[cfn_channel_budget] b
WHERE [year] = @year and [Period] = @period and [ACC] >= '30000' and cct = '0130'

UNION ALL
-- last year
SELECT [Group],cc, cct, acc, prd, local, prj, Inter,@year, period, '0' as actual, '0' as actual_ytd, [Period Activity] as actual_ly, [Ending Balance] as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year and [Period] = @period and [ACC] >= '30000' and cct = '0130'
) AS tb
LEFT JOIN (
SELECT distinct scm_retail.cost_center as [RetailCC], scm_mall.cost_center as [MallCC], scm_retail.format_name
FROM [LocalMaster].[dbo].[StoreCostCenterMapping] scm_retail
INNER JOIN (
SELECT scm_mall.storeno, scm_mall.cost_center
FROM [LocalMaster].[dbo].[StoreCostCenterMapping] scm_mall
WHERE scm_mall.cost_center_type in ('MALL CC', 'MALL CC')
) scm_mall ON scm_retail.storeno = scm_mall.storeno
WHERE scm_retail.cost_center_type in ('RETAIL CC', 'RTI CC')
) AS store on tb.cc = store.[MallCC]
LEFT JOIN [LocalMaster].[dbo].[PL_template_store_cc] s ON s.ccc = store.[RetailCC] --WHERE s.StoreFormatName <> 'New Channel'

GROUP BY tb.[Group],
tb.[ACC]
,tb.[CCT]
,tb.[PRD]
,tb.[PRJ]
,tb.[Local]
,tb.[Inter]
,tb.[Year]
,tb.[Period]
,s.cc
,s.StoreFormatName
```

ภาพที่ 3.51 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ขั้นตอนที่ 3 นำข้อมูลของรายได้ค่าใช้จ่ายที่เป็นส่วนของงานบริการที่มีข้อมูลที่ต้องการครบแล้ว ใส่ไปในตาราง TEMP\_PeriodSnapshot

```
-- STEP 3: Insert Retail Service (distribute RS to proper format (1) Change 'format' from Mall to proper-retail format (2) Change 'cc' from FR CC to Retail CC )
INSERT INTO [FIN].[dbo].[TEMP_PeriodSnapshot]
SELECT [Group],
[FIN].[dbo].[GetAccountMappingForPL] (tb.[CCT], tb.[ACC], tb.[PRD], tb.[Local]) AS [ACC]
,CASE
WHEN (tb.cc = '41000') THEN '01000'
WHEN (tb.cc = '41500') THEN '01500'
WHEN (tb.cc = '44000') THEN '04000'
WHEN (tb.cc = '45500') THEN '05500'
WHEN (tb.cc = '42000') THEN '02000'
ELSE s.[cc]
END AS [cc]
,tb.[CCT]
,tb.[PRD]
,tb.[PRD]
,tb.[Local]
,tb.[Inter]
,tb.[Year]
,tb.[Period]
,SUM(cast(tb.[actual] as numeric(18,2)) * -1) as 'Period Activity'
,SUM(cast(tb.[actual_ytd] as numeric(18,2)) * -1) as 'YTD'
,SUM(cast(tb.[actual_ly] as numeric(18,2)) * -1) as 'Period Activity LY'
,SUM(cast(tb.[actual_ly_ytd] as numeric(18,2)) * -1) as 'Period Activity LY YTD'
,SUM(cast(tb.[budget] as numeric(18,2)) * 1) as 'Period Activity TG'
,SUM(cast(tb.[budget_ytd] as numeric(18,2)) * 1) as 'Period Activity TG YTD'
,s.[StoreFormatName] AS [FormatOriginal]
,CASE
WHEN (tb.cc = '41000') THEN 'Hyper-Pool'
WHEN (tb.cc = '41500') THEN 'Extra-Pool'
WHEN (tb.cc = '42000') THEN 'Express-Pool'
WHEN (tb.cc = '44000') THEN 'Supermarket-Pool'
WHEN (tb.cc = '45500') THEN 'Department Store-Pool'
WHEN (tb.cc in ('43110', '43111')) THEN 'UNKNOWN'
ELSE [FIN].[dbo].[GetFormatMappingForPL] ([CCT], s.[cc]) : s.[StoreFormatName]
END AS [Format]
FROM
( -- this year
SELECT [Group], cc, cct, acc, prd, local, prj, Inter, [year], [period], [Period Activity] as actual, [Ending Balance] as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year and [Period] = @period and [ACC] >= '30000' and cct = '0141'

UNION ALL
-- budget
SELECT [Group], cc, cct, acc, prd, local, prj, Inter, [year], [period], '0' as actual, '0' as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, [Period Activity] as budget, [YTD] as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year and [Period] = @period and [ACC] <= '30000' and cct = '0141'

UNION ALL
-- last year
SELECT [Group], cc, cct, acc, prd, local, prj, Inter, @year as [year], [period], '0' as actual, '0' as actual_ytd, [Period Activity] as actual_ly, [Ending Balance] as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year and [Period] = @period and [ACC] <= '30000' and cct = '0141'
) AS tb
LEFT JOIN (
SELECT distinct scm_retail.cost_center as [RetailCC], scm_mall.cost_center as [MallCC], scm_retail.format_name
FROM [LocalMaster].[dbo].[StoreCostCenterMapping] scm_retail
INNER JOIN
SELECT scm_mall.storeno, scm_mall.cost_center
FROM [LocalMaster].[dbo].[StoreCostCenterMapping] scm_mall
WHERE scm_mall.cost_center_type in ('FINANCE RETAIL CC', 'FR CC')
) scm_mall ON scm_retail.storeno = scm_mall.storeno
WHERE scm_retail.cost_center_type in ('RETAIL CC', 'RT CC')
) AS store ON tb.cc = store.[MallCC]
LEFT JOIN [LocalMaster].[dbo].[PL_template_store_cc] s ON s.cct = store.[RetailCC]
GROUP BY
[Group],
tb.[Acc]
,tb.[cc]
,tb.[CCT]
,tb.[PRD]
,tb.[PRJ]
,tb.[Local]
,tb.[Inter]
,tb.[Year]
,tb.[Period]
,s.[cc]
,s.[StoreFormatName]
```

ภาพที่ 3.52 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

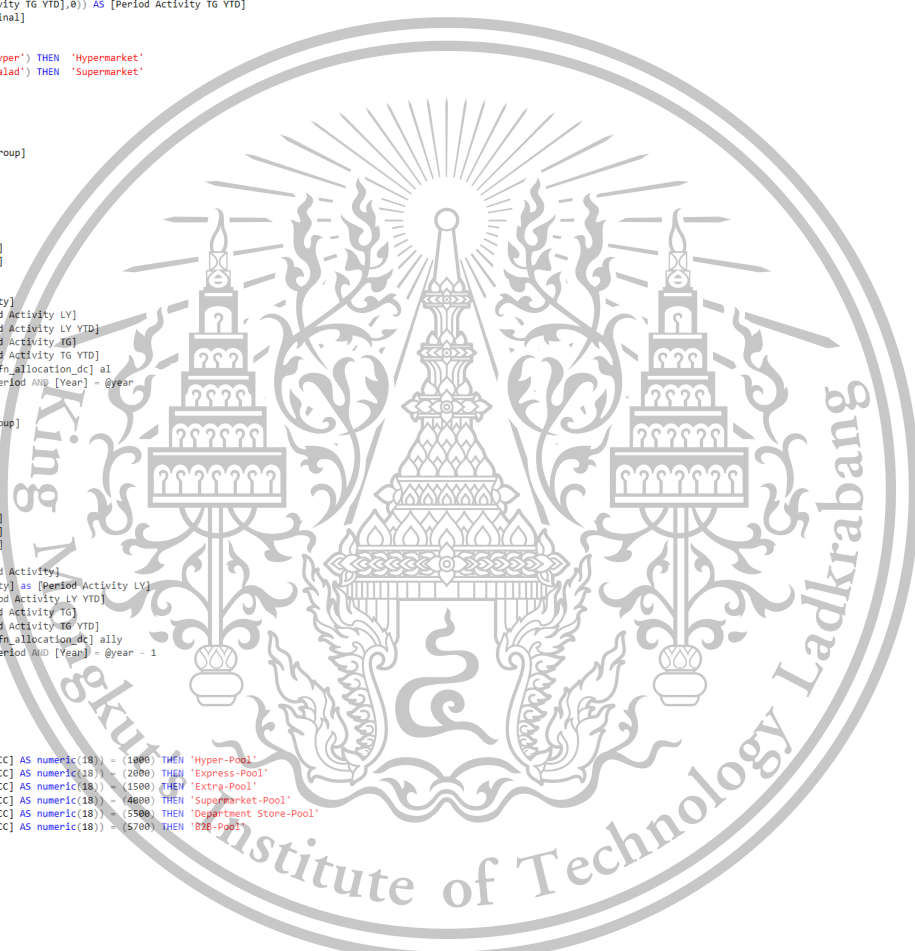
ขั้นตอนที่ 4 นำข้อมูลของรายได้ค่าใช้จ่ายที่เป็นส่วนของศูนย์กระจายสินค้าที่มีข้อมูลที่ต้องการครบแล้ว  
ใส่ไปในตาราง TEMP\_PeriodSnapshot

```
-- STEP 4: Insert DC
INSERT INTO [FIN].[dbo].[TEMP_PeriodSnapshot]
SELECT [Group]
, 'CUSTOM9901' as [Acc]
, [CC]
, '0800' AS [CCT]
, [PRD]
, [PRJ]
, [Local]
, [Inter]
, [Year]
, [Period]
, SUM(ISNULL([Period Activity],0)) as 'Period Activity'
, SUM(ISNULL([YTD],0)) as 'Period Activity YTD'
, SUM(ISNULL([Period Activity LY],0)) AS [Period Activity LY]
, SUM(ISNULL([Period Activity LY YTD],0)) AS [Period Activity LY YTD]
, SUM(ISNULL([Period Activity TG],0)) AS [Period Activity TG]
, SUM(ISNULL([Period Activity TG YTD],0)) AS [Period Activity TG YTD]
, [Format] AS [FormatOriginal]
--, [Format] AS [Format]
--, CASE
  WHEN ([Format] = 'Hyper') THEN 'Hypermarket'
  WHEN ([Format] = 'Tala') THEN 'Supermarket'
  ELSE [Format]
END AS [Format]

FROM ( SELECT '081' as [Group]
, [Name]
, [Format]
, [YTD]
, [CC]
, NULL as [PRD]
, NULL as [PRJ]
, NULL as [Local]
, NULL as [Inter]
, [Year]
, [Period]
, [Period Activity]
, NULL as [Period Activity LY]
, NULL as [Period Activity LY YTD]
, NULL as [Period Activity TG]
, NULL as [Period Activity TG YTD]
FROM [FIN].[dbo].[cfn_allocation_dc] al
where [Period] = @period AND [Year] = @year

UNION ALL
SELECT '081' as [Group]
, [Name]
, [Format]
, NULL as [YTD]
, [CC]
, NULL as [PRD]
, NULL as [PRJ]
, NULL as [Local]
, NULL as [Inter]
, @year as [Year]
, [Period]
, NULL as [Period Activity]
, [Period Activity] as [Period Activity LY]
, [YTD] as [Period Activity LY YTD]
, NULL as [Period Activity TG]
, NULL as [Period Activity TG YTD]
FROM [FIN].[dbo].[cfn_allocation_dc] ally
where [Period] = @period AND [Year] = @year - 1

UNION ALL
SELECT [Group]
, NULL as [Name]
, CASE
  WHEN CAST([CC] AS numeric(18)) = 1000 THEN 'Hyper-Pool'
  WHEN CAST([CC] AS numeric(18)) = 2000 THEN 'Express-Pool'
  WHEN CAST([CC] AS numeric(18)) = 1500 THEN 'Extra-Pool'
  WHEN CAST([CC] AS numeric(18)) = 4000 THEN 'Supermarket-Pool'
  WHEN CAST([CC] AS numeric(18)) = 5500 THEN 'Department Stone-Pool'
  WHEN CAST([CC] AS numeric(18)) = 5700 THEN 'B2B-Pool'
  ELSE NULL
END AS [Format]
, NULL as [YTD]
, [CC]
, [PRD]
, [PRJ]
, [Local]
, [Inter]
, [Year]
, [Period]
, NULL as [Period Activity]
, NULL as [Period Activity LY]
, NULL as [Period Activity LY YTD]
, [Period Activity] as [Period Activity TG]
, [YTD] as [Period Activity TG YTD]
FROM [FIN].[dbo].[fpa_channel_budget] bg
WHERE CCT = '0800' AND CAST([CC] AS numeric(18)) < (8000) AND [Period] = @period AND [Year] = @year
) as dc
GROUP BY [Group]
, [CC]
, [Year]
, [Period]
, [Format]
, [PRD]
, [PRJ]
, [Local]
, [Inter]
```



เอกสารนี้เป็นเอกสารที่ภาพที่ 3.53 ภาพแสดงโค้ดของ spPopulatePLSnapShotByPeriod (5) ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 5 นำข้อมูลของรายได้ค่าใช้จ่ายที่เป็นส่วนของสำนักงานใหญ่ที่มีข้อมูลที่ต้องการครบแล้ว ไปใส่ในตาราง TEMP\_PeriodSnapshot

```
-- STEP 5: Insert HO
INSERT INTO [FIN].[dbo].[TEMP_PeriodSnapshot]
SELECT [Group]
, 'CUSTOM9902' as [Acc]
, [CC]
, '9900' AS [CCT]
, [PRD]
, [PR2]
, [Local]
, [Inter]
, [Year]
, [Period]
, SUM(isnull([Period Activity],0)) as 'Period Activity'
, SUM(isnull([YTD],0)) as 'Period Activity YTD'
, sum(isnull([Period Activity LY],0)) as [Period Activity LY]
, sum(isnull([Period Activity LY YTD],0)) as [Period Activity LY YTD]
, SUM(isnull([Period Activity TG],0)) AS [Period Activity TG]
, SUM(isnull([Period Activity TG YTD],0)) AS [Period Activity TG YTD]
, [Format] AS [FormatOriginal]
--, [Format] AS [Format]
--, [Format] AS [Format]
, CASE
WHEN ([Format] = 'Hyper') THEN 'Hypermarket'
WHEN ([Format] = 'Talad') THEN 'Supermarket'
ELSE [Format]
END AS [Format]

FROM ( SELECT RAM
, [Name]
, [Format]
, [YTD]
, '001' as [Group]
, [CC]
, NULL as [PRD]
, NULL as [PR2]
, NULL as [Local]
, NULL as [Inter]
, [Year]
, [Period]
, [Period Activity]
, NULL as [Period Activity LY]
, NULL as [Period Activity LY YTD]
, NULL as [Period Activity TG]
, NULL as [Period Activity TG YTD]
FROM [FIN].[dbo].[cfm_allocation_ho] al
WHERE [Period] = @period AND [Year] = @year

UNION ALL -- LY
SELECT RAM
, [Name]
, [Format]
, NULL as [YTD]
, '001' as [Group]
, [CC]
, NULL as [PRD]
, NULL as [PR2]
, NULL as [Local]
, NULL as [Inter]
, @year as [Year]
, [Period]
, NULL as [Period Activity]
, [Period Activity] as [Period Activity LY]
, [YTD] as [Period Activity LY YTD]
, NULL as [Period Activity TG]
, NULL as [Period Activity TG YTD]
FROM [FIN].[dbo].[cfm_allocation_ho] ally
WHERE [Period] = @period AND [Year] = @year - 1
```

ภาพที่ 3.54 ภาพแสดงโค้ดของ spPopulatePLSnapshotByPeriod (6)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 6 นำข้อมูลของรายได้ค่าใช้จ่ายที่เป็นส่วนของออนไลน์ ร้านค้าเปิดใหม่ และอื่น ๆ ใส่ไปในตาราง TEMP\_PeriodSnapshot

```
-- Step 6: Insert HO where CC is defined i.e. HO Online, NC, B2B
INSERT INTO [FIN].[dbo].[TEMP_PeriodSnapshot]
SELECT [Group]
, [Acc]
, [CC]
, [CCT]
, [PRD]
, [PRJ]
, [Local]
, [Inter]
, [Year]
, [Period]
, SUM([Period Activity]) AS [Period Activity]
, SUM([YTD]) AS [Period Activity YTD]
, SUM([Period Activity LY]) AS [Period Activity LY]
, SUM([Period Activity LY YTD]) AS [Period Activity LY YTD]
, SUM([Period Activity Tg]) AS [Period Activity Tg]
, SUM([Period Activity Tg YTD]) AS [Period Activity Tg YTD]
, [FormatOriginal]
, [Format]
FROM
(
select tb.[Group]
, [FIN].[dbo].[GetAccountMappingForPL] (tb.[CCT], tb.[Acc], tb.[PRD], tb.[Local]) AS [Acc]
, tb.[CC]
, tb.[CCT]
, tb.[PRD]
, tb.[PRJ]
, tb.[Local]
, tb.[Inter]
, tb.[Year]
, tb.[Period]
, SUM(ISNULL(cast(tb.[actual] as numeric(18,2)),0) * -1) as [Period Activity]
, SUM(ISNULL(cast(tb.[actual_ytd] as numeric(18,2)),0) * -1) as [YTD]
, SUM(ISNULL(cast(tb.[actual_ly] as numeric(18,2)),0) * -1) as [Period Activity LY]
, SUM(ISNULL(cast(tb.[actual_ly_ytd] as numeric(18,2)),0) * -1) as [Period Activity LY YTD]
, SUM(ISNULL(cast(tb.[budget] as numeric(18,2)),0) * 1) as [Period Activity Tg]
, SUM(ISNULL(cast(tb.[budget_ytd] as numeric(18,2)),0) * 1) as [Period Activity Tg YTD]
, s.StoreFormatName AS [FormatOriginal]
, [FIN].[dbo].[GetFormatMappingForPL] (tb.[CCT], tb.[CC], s.StoreFormatName) AS [Format]
, [FIN].[dbo].[GetPoolFormatMappingForPL] (tb.[CC], s.StoreFormatName) AS [PoolFormat]
FROM
( -- this year
SELECT [Group], cc, cct, acc, prd, [local], prj, inter, [year], [period], [Period Activity] as actual, [Ending Balance] as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, '0' as budget, '0' as budget_ytd
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year and [Period] = @period and [ACC] >= '90000' and cct = '0200'
UNION ALL
-- budget
SELECT [Group], cc, cct, acc, prd, [local], prj, inter, [year], [period], '0' as actual, '0' as actual_ytd, '0' as actual_ly, '0' as actual_ly_ytd, [Period Activity] as budget, [YTD] as budget_ytd
FROM [FIN].[dbo].[fpa_channel_budget] b
WHERE [year] = @year and [Period] = @period and [ACC] >= '30000' and cct = '0900'
UNION ALL
-- last year
SELECT [Group], cc, cct, acc, prd, [local], prj, inter, @year as [year], [period], '0' as actual, '0' as actual_ytd, [Period Activity] as actual_ly, [Ending Balance] as actual_ly_ytd, '0' as budget, '0' as budget_y
FROM [FIN].[dbo].[cfn_trial_balance_detail_A] tb
WHERE [year] = @year - 1 and [Period] = @period and [ACC] >= '30000' and cct = '0900'
) as tb
LEFT JOIN [LocalMaster].[dbo].[PL_template_store_cc] s ON s.cc = tb.cc
GROUP BY tb.[Group]
, tb.[Acc]
, tb.[CC]
, tb.[CCT]
, tb.[PRD]
, tb.[PRJ]
, tb.[Local]
, tb.[Inter]
, tb.[Year]
, tb.[Period]
, s.StoreFormatName
) as tb
WHERE [CC] IN ('09880', '09881', '09240', '09000', '09220')
GROUP BY [Group]
, [Acc]
, [CC]
, [CCT]
, [PRD]
, [PRJ]
, [Local]
, [Inter]
, [Year]
, [Period]
, [FormatOriginal]
, [Format]
PRINT N'Part 1: Mapping Account/Format and prepare data **complete**'
END
GO
```

ภาพที่ 3.55 ภาพแสดงโค้ดของ spPopulatePLSnapshotByPeriod (7)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- spLoadTBToPLSnapshot เป็น Stored procedure ที่สร้างขึ้นเพื่อนำข้อมูลที่ถูเก็บอยู่ในตาราง TEMP\_PeriodSnapshot ย้ายไปเก็บที่ตาราง fpa\_channel\_pl\_snapshot

Stored procedure นี้มีการรับพารามิเตอร์ปีและเดือน เพื่อกำหนดข้อมูลที่ต้องการนำเข้าตาราง ขั้นตอนการย้ายข้อมูลตรงนี้ไม่ได้มีการแปลงค่าข้อมูลใด ๆ โดยหลังจากที่เก็บข้อมูลเข้าตารางใหม่แล้ว โปรแกรมทำการลบตาราง TEMP\_PeriodSnapshot ทิ้งทันที

```
-- -----  
-- Author: <Haruethai A.>  
-- Create date: <Sep 4, 2020>  
-- Description: <This view loads trial balance of the specified {year,period} into the P&L-ready format>  
-- -----  
ALTER PROCEDURE [dbo].[spLoadTBToPLSnapshot] (@year as integer = NULL, @period as integer = NULL)  
AS  
BEGIN  
-- SET NOCOUNT ON added to prevent extra result sets from  
-- interfering with SELECT statements.  
SET NOCOUNT ON;  
  
-- Insert statements for procedure here  
INSERT INTO [FIN].[dbo].[fpa_channel_pl_snapshot]  
SELECT  
[Group],  
[Acc],  
[CC],  
[CCT],  
[PRD],  
[PRJ],  
[Local],  
[Inter],  
[Year],  
[Period],  
[Period Activity] AS [Period Activity],  
[Period Activity YTD] AS [Period Activity YTD],  
[Period Activity LY] AS [Period Activity LY],  
[Period Activity LY YTD] AS [Period Activity LY YTD],  
[Period Activity TG] AS [Period Activity TG],  
[Period Activity TG YTD] AS [Period Activity TG YTD],  
[FormatOriginal],  
[Format]  
FROM [FIN].[dbo].[TEMP_PeriodSnapshot];  
DROP TABLE [FIN].[dbo].[TEMP_PeriodSnapshot];  
PRINT N'Part 2: Create PL snapshot **complete**'  
END  
GO
```

ภาพที่ 3.56 ภาพแสดงโค้ดของ spLoadTBToPLSnapshot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- spAllocateTBBByPeriod เป็น Stored procedure ที่สร้างขึ้นเพื่อเรียกใช้งาน Stored procedure อื่นที่ทำการกระจายข้อมูล โดยมีการเรียกใช้ 5 ครั้ง แบ่งตามประเภทของร้านค้า

Stored procedure นี้มีการรับพารามิเตอร์ปีและเดือน เพื่อกำหนดข้อมูลที่ต้องการ ลบข้อมูลที่อยู่ในตาราง fpa\_channel\_pl\_allocate\_snapshot ออกเฉพาะปีและเดือนนั้น ๆ จากนั้นทำการเรียกใช้งาน spAllocateTBBByPeriodByFormat โดยแต่ละครั้งที่เรียกใช้งานจะส่งค่าปี เดือน และประเภทของร้านค้าเข้าไปสุดท้ายทำการแก้ไขค่าที่ถูกลบค่าไปกระจายลงในแต่ละร้านค้าเรียบร้อยแล้วให้เป็นศูนย์ทั้งหมด เพื่อให้ได้ผลรวมของข้อมูลเท่ากับข้อมูลก่อนการกระจาย

```
-- -----  
-- Author: <Haruethai A.>  
-- Create date: <Sep 4, 2020>  
-- Update date: <Oct 22, 2020> <delete extinct data and run using new data>  
-- Description: <This store procedure allocate trial balance data for 5 retail format>  
-- Parameters: @year, @period  
  
-- EXEC [dbo].[spAllocateTBBByPeriod] @year='2021', @period='05'  
  
-- -----  
ALTER PROCEDURE [dbo].[spAllocateTBBByPeriod] (@year as varchar(4), @period as varchar(2))  
AS  
BEGIN  
-- SET NOCOUNT ON added to prevent extra result sets from  
-- interfering with SELECT statements.  
SET NOCOUNT ON;  
  
DELETE [FIN].[dbo].[fpa_channel_pl_allocate_snapshot] where [Year] = @year and [Period] = @period  
  
exec [dbo].[spAllocateTBBByPeriodByFormat] @year=@year, @period=@period, @format='Supermarket', @pool='Supermarket-Pool'  
exec [dbo].[spAllocateTBBByPeriodByFormat] @year=@year, @period=@period, @format='Department Store', @pool='Department Store-Pool'  
exec [dbo].[spAllocateTBBByPeriodByFormat] @year=@year, @period=@period, @format='Hypermarket', @pool='Hyper-Pool'  
exec [dbo].[spAllocateTBBByPeriodByFormat] @year=@year, @period=@period, @format='Extra', @pool='Extra-Pool'  
exec [dbo].[spAllocateTBBByPeriodByFormat] @year=@year, @period=@period, @format='Express', @pool='Express-Pool'  
  
UPDATE [FIN].[dbo].[fpa_channel_pl_allocate_snapshot]  
SET [Period_Activity] = '0', [Period_Activity_YTD] = '0', [Period_Activity_LY] = '0', [Period_Activity_LY_YTD] = '0'  
WHERE [Format] IN ('Supermarket-Pool', 'Department Store-Pool', 'Hyper-Pool', 'Extra-Pool', 'Express-Pool')  
AND Acc IN (select Account_Code2 as [Account_Code]  
from [LOCALMASTER].[dbo].[PL_template_store]  
where [Z.L_PL_Line] in ('Store Payroll', 'Retail Expenses - Store'))  
AND CC IN ('01000', '01500', '05500', '04000', '02000')  
AND [Year] = @year AND [Period] = @period  
  
PRINT N'Part 3: Create PL allocation snapshot **complete**'  
  
END  
GO
```

ภาพที่ 3.57 ภาพแสดงโค้ดของ spAllocateTBBByPeriod

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- spAllocateTBBByPeriodByFormat เป็น Stored procedure ที่สร้างขึ้นเพื่อนำเอาข้อมูลรายได้และค่าใช้จ่าย ส่วนกลางของแต่ละประเภทของร้านค้ากระจายเข้าไปให้กับร้านค้าที่อยู่ภายใต้ประเภทของร้านค้านั้น ๆ โดยกระจายตามเปอร์เซ็นต์ยอดขายของร้านค้านั้น ๆ

Stored procedure นี้มีการรับพารามิเตอร์ปี เดือน และประเภทของร้านค้า เตรียมตารางที่จะนำมาใช้ในการเก็บข้อมูลชั่วคราว

```

-- -----
-- Author: <Haruethai A.>
-- Create date: <Sep 4, 2020>
-- Description: <This store procedure allocate trial balance data from pool to store (allocate only Store Payroll and Retail Expenses - Store)>
-- Parameters: @year, @period, @format, @pool

-- exec [dbo].[spAllocateTBBByPeriodByFormat_v2] @year='2021', @period='05', @format='Express', @pool='Express-Pool'

-- -----
ALTER PROCEDURE [dbo].[spAllocateTBBByPeriodByFormat] (@year as varchar(4), @period as varchar(2), @format as varchar(25), @pool as varchar(25))
AS
BEGIN
-- SET NOCOUNT ON added to prevent extra result sets from
-- interfering with SELECT statements.
SET NOCOUNT ON;

--clear temp table
IF OBJECT_ID('[FIN].[dbo].[temp_pool_expense]') IS NOT NULL DROP TABLE [FIN].[dbo].[temp_pool_expense]
IF OBJECT_ID('[FIN].[dbo].[temp_sale_store]') IS NOT NULL DROP TABLE [FIN].[dbo].[temp_sale_store]
IF OBJECT_ID('[FIN].[dbo].[temp_allocate]') IS NOT NULL DROP TABLE [FIN].[dbo].[temp_allocate]

-- STEP 1: GET POOL EXPENSES
SELECT
[Group],
Acc,
CC,
CCT,
PRD,
--PRJ,
[Local],
--Inter,
SUM(Period_Activity) AS PA,
SUM(Period_Activity_YTD) AS PA_YTD,
SUM(Period_Activity_LY) AS PA_LY,
SUM(Period_Activity_LY_YTD) AS PA_LY_YTD,
SUM(Period_Activity_TG) AS PA_TG,
SUM(Period_Activity_TG_YTD) AS PA_TG_YTD,
FormatOriginal,
[Format]
INTO [FIN].[dbo].[temp_pool_expense]
FROM [FIN].[dbo].[fpa_channel_pl_snapshot]
WHERE [year] = @year AND [period] = @period AND [Format] = @pool -- 'Supermarket-Pool', 'Department Store-Pool', 'Hyper-Pool', 'Extra-Pool', 'Express-Pool'
AND Acc IN (select Account_Code2 as [Account_Code] from [LOCALMASTER].[dbo].[PL_template_store] where [2:1:PL_Line] in ('Store Payroll','Retail Expenses - Store'))
AND CC IN ('01900','01500','05500','04000','02000')
AND ([Period_Activity] != 0 OR [Period_Activity_YTD] != 0 OR [Period_Activity_LY] != 0 OR [Period_Activity_LY_YTD] != 0)
GROUP BY
[Group],
Acc,
CC,
CCT,
PRD,
--PRJ,
[Local],
--Inter,
FormatOriginal,
[Format]

```

ภาพที่ 3.58 ภาพแสดงโค้ดของ spAllocateTBBByPeriodByFormat (1)  
 ขั้นตอนที่ 1 กรองข้อมูลให้เหลือเฉพาะข้อมูลที่เป็นรายได้และค่าใช้จ่ายส่วนกลาง

ภาพที่ 3.59 ภาพแสดงโค้ดของ spAllocateTBBByPeriodByFormat (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ขั้นตอนที่ 2 คำนวณเปอร์เซ็นต์ยอดขายของแต่ละร้านค้า

```
-- STEP 2: CALCULATE % TO SALE BY STORE
select
sale_by_store.[Format] as [Format],
sale_by_store.[CC] as CC,
--% to sales Period Activity
sale_by_store.sale_PA AS [Sale_PA_ByStore],
sale_by_format.sale_PA AS [Sale_PA_ByFormat],
sale_by_store.sale_PA / sale_by_format.sale_PA AS [PercentSale_PA_ByStore],
--% to sales YTD
sale_by_store.sale_YTD AS [Sale_YTD_ByStore],
sale_by_format.sale_YTD AS [Sale_YTD_ByFormat],
sale_by_store.sale_YTD / sale_by_format.sale_YTD AS [PercentSale_YTD_ByStore],
--% to sales Period Activity LY
sale_by_store.sale_LY AS [Sale_LY_ByStore],
sale_by_format.sale_LY AS [Sale_LY_ByFormat],
sale_by_store.sale_LY / sale_by_format.sale_LY AS [PercentSale_LY_ByStore],
--% to sales Period Activity LY YTD
sale_by_store.sale_LY_YTD AS [Sale_LY_YTD_ByStore],
sale_by_format.sale_LY_YTD AS [Sale_LY_YTD_ByFormat],
sale_by_store.sale_LY_YTD / sale_by_format.sale_LY_YTD AS [PercentSale_LY_YTD_ByStore]
into [FIN].[dbo].temp_sale_store

FROM
-- PART 1: sale by store
((SELECT --*
[Format]
,CC
,SUM([Period_Activity]) AS sale_PA
,SUM([Period_Activity_YTD]) AS sale_YTD
,SUM([Period_Activity_LY]) AS sale_LY
,SUM([Period_Activity_LY_YTD]) AS sale_LY_YTD
FROM [FIN].[dbo].[fpa_channel_pl_snapshot]
WHERE [year] = @year AND [period] = @period AND [Format] = @format
AND Acc IN (select Account_Code2 as [Account_Code] from [LOCALMASTER].[dbo].[PL_template_store] where [2_PL_Line] = 'Net Sale excl.VAT')
AND CC not between '85700' and '85997')
GROUP BY [Format], Acc, CC, CCT, PRD, [Local]
) as sale_by_store

LEFT JOIN
-- PART 2: sale by format
(SELECT --*
[Format]
--SUM([Period_Activity]) AS sale
,SUM([Period_Activity]) AS sale_PA
,SUM([Period_Activity_YTD]) AS sale_YTD
,SUM([Period_Activity_LY]) AS sale_LY
,SUM([Period_Activity_LY_YTD]) AS sale_LY_YTD
FROM [FIN].[dbo].[fpa_channel_pl_snapshot]
WHERE [year] = @year AND [period] = @period AND [Format] = @format
AND Acc IN (select Account_Code2 as [Account_Code] from [LOCALMASTER].[dbo].[PL_template_store] where [2_PL_Line] = 'Net Sale excl.VAT')
AND CC not between '85700' and '85997')
GROUP BY [Format]
) as sale_by_format
ON sale_by_store.[Format] = sale_by_format.[Format])
```

ภาพที่ 3.60 ภาพแสดงโค้ดของ spAllocateTBBByPeriodByFormat (3)

## ขั้นตอนที่ 3 ทำการกระจายรายได้และค่าใช้จ่ายส่วนกลางไปที่ร้านค้า

```
-- STEP 3: PREPARE CC ACCOUNT TO ALLOCATE FROM POOL
-- ALLOCATE TO STORE USING % SALE
select template_cc.cc as CC, template_cc.acc as Acc, template_cc.cct as CCT--, template_cc.prd as PRD, template_cc.local as [Local]
, (template_cc.Pool_PA * s.PercentSale_PA_ByStore) as [Period_Activity]
, (template_cc.Pool_PA_YTD * s.PercentSale_YTD_ByStore) as [Period_Activity_YTD]
, (template_cc.Pool_LY * s.PercentSale_LY_ByStore) as [Period_Activity_LY]
, (template_cc.Pool_LY_YTD * s.PercentSale_LY_YTD_ByStore) as [Period_Activity_LY_YTD]
INTO [FIN].[dbo].temp_allocate
from
(select cc.cc, pool.acc, pool.cct
, p.PA as [Pool_PA], p.PA_YTD as [Pool_PA_YTD], p.PA_LY as [Pool_LY], p.PA_LY_YTD as [Pool_LY_YTD]
from
(select distinct cc from [FIN].[dbo].temp_sale_store) cc -- store to be allocated
, (select distinct acc, cct from [FIN].[dbo].temp_pool_expense where [Local] != '80976') pool -- (acc, prd, local) to be allocated
left join [FIN].[dbo].temp_pool_expense p on pool.acc = p.acc --and pool.prd = p.prd and pool.local = p.local
) template_cc
left join [FIN].[dbo].temp_sale_store s on template_cc.cc = s.cc
```

ภาพที่ 3.61 ภาพแสดงโค้ดของ spAllocateTBBByPeriodByFormat (4)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



ขั้นตอนที่ 4 นำข้อมูลเข้าไปเก็บไว้ที่ตาราง fpa\_channel\_pl\_allocate\_snapshot โดยนอกจากจะนำข้อมูลในส่วนที่ได้ทำการกระจายรายได้และค่าใช้จ่ายส่วนกลางเข้าไปเก็บในตารางแล้วนั้น ยังนำข้อมูลที่ไม่ถูกกระจายเข้าไปเก็บไว้ด้วย

```

-- STEP 4: LOAD ORIGINAL TB TO DESTINATION TABLE
--trial balance
insert into [FIN].[dbo].fpa_channel_pl_allocate_snapshot
SELECT
    [Group],
    Acc,
    CC,
    CCT,
    PRD,
    PRJ,
    [Local],
    Inter,
    [Year],
    [Period],
    Period_Activity AS [Period_Activity],
    Period_Activity_YTD AS [Period_Activity_YTD],
    Period_Activity_LY AS [Period_Activity_LY],
    Period_Activity_LY_YTD AS [Period_Activity_LY_YTD],
    Period_Activity_TG AS [Period_Activity_TG],
    Period_Activity_TG_YTD AS [Period_Activity_TG_YTD],
    [Format]
FROM [FIN].[dbo].[fpa_channel_pl_snapshot] s
WHERE [year] = @year AND [period] = @period AND [Format] in (@format, @pool)

-- allocated
insert into [FIN].[dbo].fpa_channel_pl_allocate_snapshot
SELECT
    '081' as [Group],
    Acc,
    CC,
    CCT,
    '000' as PRD,
    '00000' as PRJ,
    '00000' as [Local],
    '000' as Inter,
    @year as [Year],
    @period as [Period],
    [Period_Activity],
    [Period_Activity_YTD],
    [Period_Activity_LY],
    [Period_Activity_LY_YTD],
    '0' AS [Period_Activity_TG],
    '0' AS [Period_Activity_TG_YTD],
    --FormatOriginal,
    @format as [Format]
--INTO [FIN].[dbo].pl_allocate_test
FROM [FIN].[dbo].temp_allocate
WHERE ([Period_Activity] != 0 OR [Period_Activity_YTD] != 0 OR [Period_Activity_LY] != 0 OR [Period_Activity_LY_YTD] != 0)

/*UPDATE [FIN].[dbo].fpa_channel_pl_allocate_snapshot
SET [Period_Activity] = '0', [Period_Activity_YTD] = '0', [Period_Activity_LY] = '0', [Period_Activity_LY_YTD] = '0'
--WHERE [Format] IN ('Supermarket-Pool', 'Department-Store-Pool', 'Hyper-Pool', 'Extra-Pool', 'Express-Pool')
WHERE [Format] = @pool
and Acc IN (select Account_Code2 as [Account_Code] from [LOCALMASTER].[dbo].[PL_template_store] where [2.1]PL_Line in ('Store Payroll', 'Retail Expenses - Store'))
AND CC in ('01000', '01500', '05500', '04000', '02000')*/

-- Clean up
drop table [FIN].[dbo].temp_pool_expense
drop table [FIN].[dbo].temp_sale_store
drop table [FIN].[dbo].temp_allocate
END
GO

```

ภาพที่ 3.62 ภาพแสดงโค้ดของ spAllocateTBBByPeriodByFormat (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) สร้าง View เพื่อให้โปรแกรม Microsoft Power BI สามารถนำไปใช้ได้อย่างรวดเร็ว

- fpa\_channel\_pl\_snapshot\_v นำข้อมูลจากตารางข้อมูลรายได้ค่าใช้จ่าย ซึ่งข้อมูลถูกแปลงเสร็จเรียบร้อยแล้ว นำข้อมูลมาหารด้วยหนึ่งพัน เพื่อให้ข้อมูลอยู่ในหน่วยพันบาท

```
-- -----  
-- Author: <Haruethai A.>  
-- Create date: <Sep 4, 2020>  
-- Description: <This view use data from snapshot table, divide data with 1000>  
  
-- EXEC [FIN].[dbo].[spTBGetYearPeriod]  
-- -----
```

```
ALTER view [dbo].[fpa_channel_pl_snapshot_v] as  
SELECT [Group]  
, [Acc]  
, [CC]  
, [CCT]  
, [PRD]  
, [PRJ]  
, [Local]  
, [Inter]  
, [Year]  
, [Period]  
, [Period_Activity] /1000 AS [Period_Activity]  
, [Period_Activity_YTD] /1000 AS [Period_Activity_YTD]  
, [Period_Activity_LY] /1000 AS [Period_Activity_LY]  
, [Period_Activity_LY_YTD] /1000 AS [Period_Activity_LY_YTD]  
, [Period_Activity_TG] /1000 AS [Period_Activity_TG]  
, [Period_Activity_TG_YTD] /1000 AS [Period_Activity_TG_YTD]  
, [FormatOriginal]  
, [Format]  
FROM [FIN].[dbo].[fpa_channel_pl_snapshot]  
GO
```

ภาพที่ 3.63 ภาพแสดงโค้ดของ fpa\_channel\_pl\_snapshot\_v

- fpa\_channel\_pl\_allocate\_snapshot\_v นำข้อมูลจากตารางข้อมูลรายได้ค่าใช้จ่ายที่มีการกระจายรายได้และค่าใช้จ่ายส่วนกลางไปยังร้านค้าปลีก ซึ่งข้อมูลถูกแปลงเสร็จเรียบร้อยแล้ว นำข้อมูลมาหารด้วยหนึ่งพัน เพื่อให้ข้อมูลอยู่ในหน่วยพันบาท

```
-- -----  
-- Author: <Haruethai A.>  
-- Create date: <Sep 4, 2020>  
-- Description: <This view use data from snapshot table, divide data with 1000>  
  
-- EXEC [FIN].[dbo].[spTBGetYearPeriod]  
-- -----
```

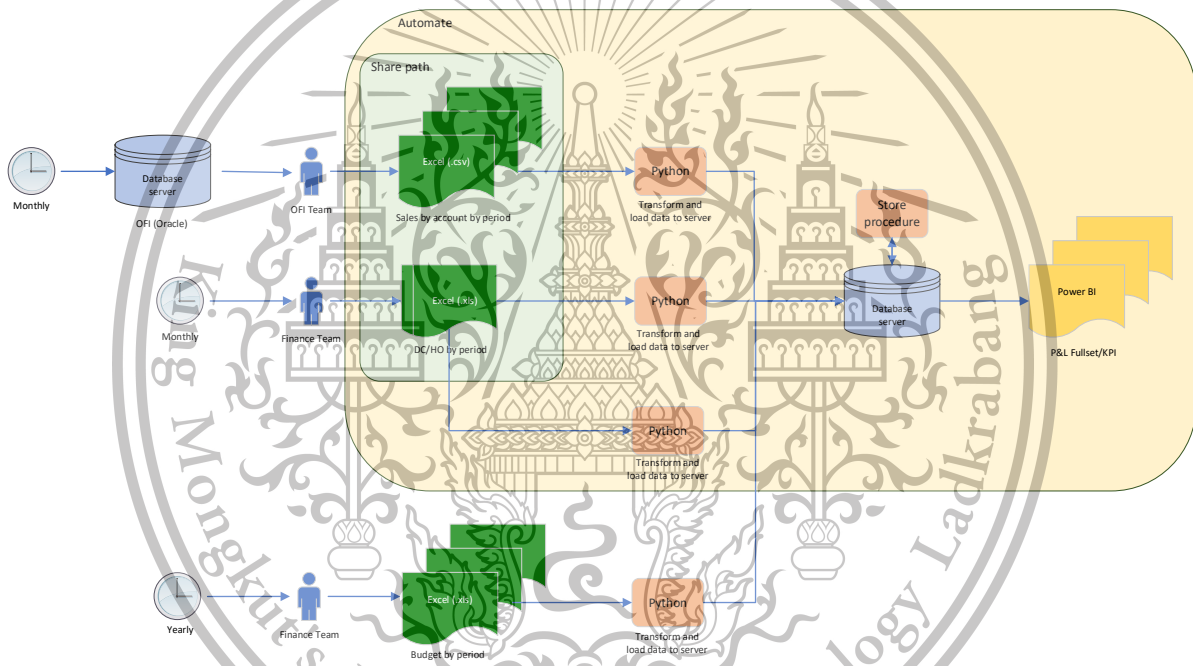
```
ALTER VIEW [dbo].[fpa_channel_pl_allocate_snapshot_v] AS  
SELECT [Group]  
, Acc  
, CC  
, CCT  
, PRD  
, PRJ  
, [Local]  
, Inter  
, [Year]  
, [Period]  
, [Period_Activity]/1000 as [Period_Activity]  
, [Period_Activity_YTD]/1000 as [Period_Activity_YTD]  
, [Period_Activity_LY]/1000 as [Period_Activity_LY]  
, [Period_Activity_LY_YTD]/1000 as [Period_Activity_LY_YTD]  
, [Period_Activity_TG]/1000 as [Period_Activity_TG]  
, [Period_Activity_TG_YTD]/1000 as [Period_Activity_TG_YTD]  
, [Format]  
FROM [dbo].[fpa_channel_pl_allocate_snapshot]  
GO
```

ภาพที่ 3.64 ภาพแสดงโค้ดของ fpa\_channel\_pl\_allocate\_snapshot\_v

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น ยกเว้นแต่กรณีที่มีการขออนุญาตจากทางสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

### 3.4.2 ออกแบบส่วนอัตโนมัติ

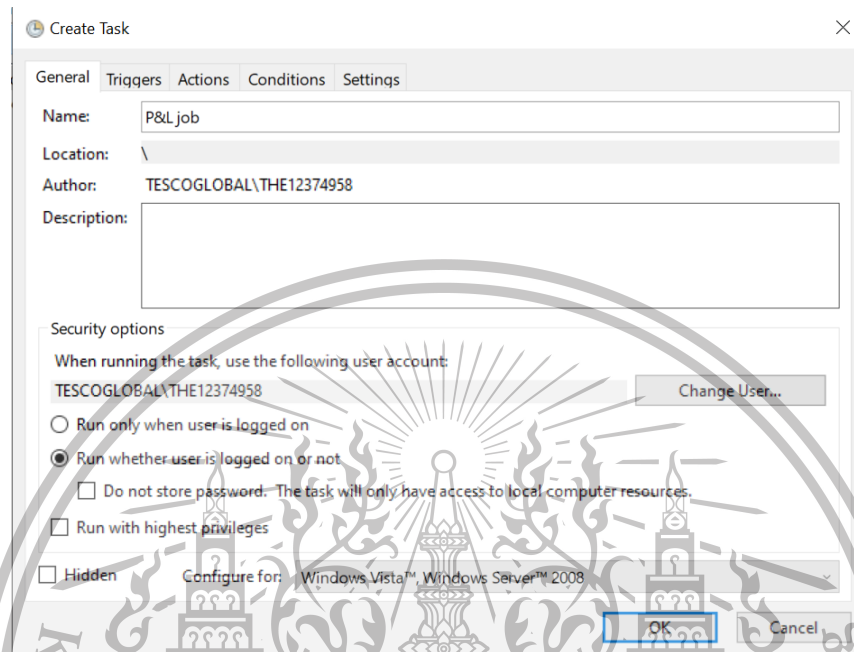
กระบวนการไหลข้อมูลที่ได้ออกแบบและพัฒนานั้นจะพัฒนาทำส่วนที่เป็นแบบอัตโนมัติ โดยส่วนของอัตโนมัติจะครอบคลุมตั้งแต่การจัดการข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริง ข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริงที่ศูนย์กระจายสินค้า และข้อมูลรายได้และค่าใช้จ่ายที่เกิดขึ้นจริงที่สำนักงานใหญ่ ในโปรแกรม Python โหลดข้อมูลเข้าฐานข้อมูล เรียกใช้งาน Stored Procedure และอัปเดตข้อมูลที่แสดงในรายงานงบกำไรขาดทุนที่โปรแกรม Microsoft Power BI ให้เป็นข้อมูลล่าสุด ที่ไม่ทำส่วนของข้อมูลรายได้และค่าใช้จ่ายที่คาดว่าจะเกิดขึ้นนั้น เนื่องจากในส่วนนี้มีข้อมูลเพิ่มเข้ามาเพียงปีละครั้งเท่านั้น และรูปแบบของข้อมูลที่ได้มาไม่ตายตัวทำให้จัดทำให้เป็นอัตโนมัติไม่ได้



ภาพที่ 3.65 ภาพแสดงกระบวนการที่ครอบคลุมในส่วนของอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการของส่วนอัตโนมัติเริ่มจากใช้โปรแกรม Task Scheduler ตั้งค่าโปรแกรมให้ไปเรียกใช้งานไฟล์ Python ตั้งให้ทำงานทุกวันศุกร์ เวลา 19 นาฬิกาตรง



ภาพที่ 3.66 ภาพแสดงการตั้งค่าโปรแกรม Task Scheduler

ที่ไฟล์ Python เพิ่มและแก้ไขโค้ดเพื่อให้สามารถทำงานอัตโนมัติได้ มีการเพิ่มในส่วนของการตรวจสอบว่ามีไฟล์ของเดือนนั้น ๆ ดูกว้างไว้ในไฟล์เตอร์ส่วนกลางหรือไม่ และเก็บชื่อไฟล์เอาไว้

```
def check_files():
    path = r'\\172.30.127.109\Reporting\PL'
    files = os.listdir(path)
    files_list = []
    prep_files = []
    for f in files:
        result_list = re.findall(r'OFi_GL_PL_(.*)', f)
        files_list.append(result_list)
    prep_files = [x for x in files_list if x != []]
    if len(prepare_files) > 0:
        return prep_files, path

def prep_files():
    try:
        files = check_files()[0]
        no_of_files = len(files)
        remove_bracket = []
        staging_files = []
        for j in range(no_of_files):
            remove_bracket.append(files[j])
        data_a = [i[0] for i in remove_bracket]
        data_b = []
        for i in data_a:
            data_b.append('OFi_GL_PL_'+i)
        return data_b
    except:
        query_job_file_not_exist()
        sys.exit
```

ภาพที่ 3.67 ภาพแสดงโค้ด Python ในส่วนอัตโนมัติ (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น โปรดอย่าได้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## ตรวจสอบรูปแบบของข้อมูล แปลงข้อมูล และโหลดข้อมูลเข้าฐานข้อมูล

```
def check_valid_content():
    engine = init_mssql_connection()
    start_time = time.time()
    try:
        path = check_files()[1]
        source_name = create_key()[1]
        source = path+'\\'+source_name+'.txt'
    except:
        query_job_file_not_exist()
        sys.exit()
    df = pd.read_csv(source, delimiter="|", header=None, dtype = str)
    query_job_content_valid(df)
    df.columns = ["Period-Year", "Group", "Acc", "CC", "CCT", "PRD", "PRJ", 'Local', 'Inter', 'Beginning Balance', 'Period Activity',
    'Ending Balance', 'Currency']
    df['Period'] = df['Period-Year'].str.slice(0,2)
    df['Year'] = '20' + df['Period-Year'].str.slice(3,5)
    df = df[['Group', 'Acc', 'CC', 'CCT', 'PRD', 'PRJ', 'Local', 'Inter', 'Year', 'Period', 'Beginning Balance', 'Period Activity', 'Ending
    Balance']]
    df[["Beginning Balance", "Ending Balance", "Period Activity"]] = df[["Beginning Balance", "Ending Balance", "Period
    Activity"]].astype(float)
    total1 = int(df["Beginning Balance"].sum())
    total2 = int(df["Period Activity"].sum())
    total3 = int(df["Ending Balance"].sum())
    try:
        if (total1 + total2) - total3 == 0:
            DEST_TABLE = f'{create_key()[3]}'
            method = 'append'
            df.to_sql(DEST_TABLE, engine, if_exists=method, index=False, schema="dbo", chunksize = 50000,
            dtype = {'Group':sa.types.VARCHAR(length=50),
            'Acc':sa.types.VARCHAR(length=50),
            'CC':sa.types.VARCHAR(length=50),
            'CCT':sa.types.VARCHAR(length=50),
            'PRD':sa.types.VARCHAR(length=50),
            'PRJ':sa.types.VARCHAR(length=50),
            'Local':sa.types.VARCHAR(length=50),
            'Inter':sa.types.VARCHAR(length=50),
            'Year':sa.types.VARCHAR(length=50),
            'Period':sa.types.VARCHAR(length=50),
            'Beginning Balance':sa.types.Float,
            'Period Activity':sa.types.Float,
            'Ending Balance':sa.types.Float})
            year = df['Year'].head(1).values[0][:4]
            period = df['Period'].head(1).values[0]
            query_job_row_compare(year, period)
            call_store_procedure(year, period)
```

ภาพที่ 3.68 ภาพแสดงโค้ด Python ในส่วนอัตโนมัติ (2)

หลังจากโหลดข้อมูลเข้าฐานข้อมูลเรียบร้อยแล้ว สิ่งเรียกใช้งาน Stored Procedure ดังนี้

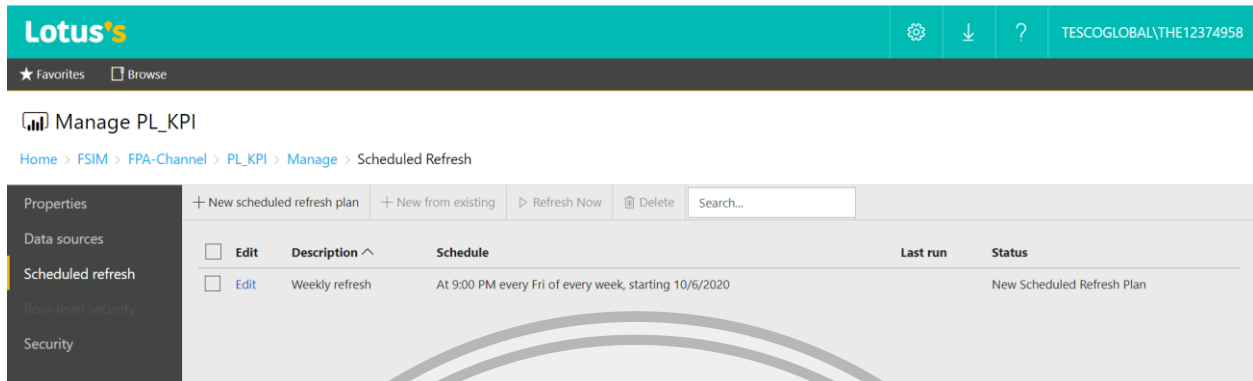
```
def call_store_procedure(year, period):
    query1 = text(f'''EXEC [FIN].[dbo].[sp]ByPeriod @year = '{year}', @period = '{period}' ''')
    query2 = text(f'''SELECT [count_matched] FROM [FIN].[dbo].[job] WHERE [key] = '{create_key()[0]}' ''')
    engine = init_mssql_connection()
    connection = engine.connect()
    trans = connection.begin()
    try:
        result = connection.execute(query2)
        trans.commit()
    except:
        trans.rollback()
        raise

    final_result = result.first()[0]
    if final_result == True:
        trans2 = connection.begin()
        result2 = connection.execute(query1)
        trans2.commit()
    else:
        sys.exit
```

ภาพที่ 3.69 ภาพแสดงโค้ด Python ในส่วนอัตโนมัติ (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของหน้า dashboard มีการตั้งให้โหลดข้อมูลใหม่เข้ารายงานทุกวันศุกร์ เวลา 21 นาฬิกาตรง



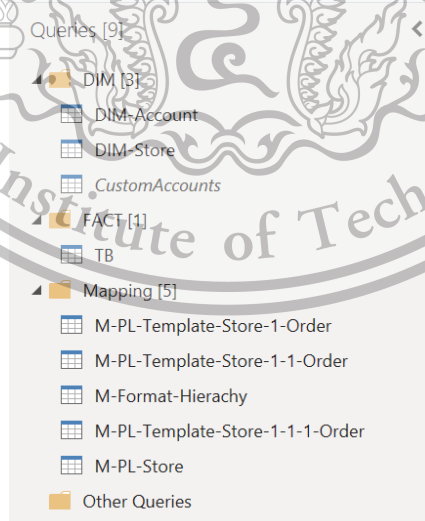
ภาพที่ 3.70 ภาพแสดงการตั้งค่าใน Microsoft Power BI Server

การทำส่วนของอัตโนมัตินั้นจะดำเนินการที่เครื่องเซิร์ฟเวอร์ ไฟล์ที่จำเป็นทั้งหมดจะถูกเก็บในเครื่องเซิร์ฟเวอร์ เพื่อให้ง่ายต่อการเรียกใช้งาน

### 3.5 สร้างหน้ารายงาน

#### 3.5.1 โหลดข้อมูลเข้า Microsoft Power BI

หลังจากที่เตรียมข้อมูลในฐานข้อมูลเสร็จแล้ว เชื่อมต่อ Microsoft Power BI กับฐานข้อมูล ดึงข้อมูลเข้า Microsoft Power BI ทั้งข้อมูลรายได้ ค่าใช้จ่าย และข้อมูล master ต่าง ๆ โดยทำการนำเข้าข้อมูลแบบ Import คือข้อมูลจะไม่รีเฟรชทุกครั้งที่เราใช้งาน



ภาพที่ 3.71 ภาพแสดงตารางที่นำเข้า Microsoft Power BI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

1) ตาราง CustomAccounts เป็นตารางที่เก็บรายละเอียดและคำนิยามของรายการรายได้ค่าใช้จ่ายที่นอกเหนือจากรายการปกติ

- นำเข้าข้อมูลมาจากรฐานข้อมูล FINMASTER จากตาราง fpa-channel\_pl\_custom\_account\_master



```
let
Source = Sql.Database("THTVDRASQLH001", "FINMASTER"),
#"dbo_fpa-channel_pl_custom_account_master" = Source[[Schema="dbo",Item="fpa-channel_pl_custom_account_master"]][Data]
in
#"dbo_fpa-channel_pl_custom_account_master"
```

✓ No syntax errors have been detected.

Done Cancel

ภาพที่ 3.72 ภาพแสดงการนำเข้าข้อมูลของตาราง CustomAccounts

2) ตาราง DIM-Account เป็นตารางที่เก็บรายละเอียดและคำนิยามของรายการรายได้ค่าใช้จ่าย

- นำเข้าข้อมูลมาจากรฐานข้อมูล FINMASTER จากตาราง account\_master

- มีการนำเอาข้อมูลจากรตาราง CustomAccounts เข้ามาเพิ่มต่อจากข้อมูลเดิมที่มีอยู่

- มีการสร้างคอลัมน์เพิ่มชื่อ AccName สร้างไคเนการนำคอดั้ม Accountcode กับคอดั้ม Description มาต่อกันคั้่นด้วยเครื่องหมาย “:”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



ภาพที่ 3.73 ภาพแสดงการนำเข้าข้อมูลของตาราง DIM-Account

- 3) ตาราง DIM-Store เป็นตารางที่เก็บข้อมูลรายละเอียดของร้านค้าปลีก เช่น เลขที่ของร้าน ชื่อร้าน ชื่อผู้จัดการ และอื่น ๆ
- นำเข้าข้อมูลมาจากรฐานข้อมูล LOCALMASTER จากตาราง PL\_template\_store\_cc
  - ข้อมูลที่นำเข้ามาจะกรองข้อมูลแถวที่มีคอลัมน์ CC เป็นค่าว่างออก
  - กรองข้อมูลแถวที่มีคอลัมน์ StoreFormatName เป็นชื่อ New Channel ออก
  - สร้างคอลัมน์เพิ่มชื่อ StoreCC สร้างโหนดการนำคอลัมน์ CC กับคอลัมน์ StoreName มาต่อกันด้วย “:”
  - เปลี่ยนประเภทข้อมูลของคอลัมน์ CC ให้เป็น Int

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



## DIM-Store

Display Options ?

```

let
Source = Sql.Database("THTVDRASQLH001", "LOCALMASTER"),
#"dbo_fpa-channel_store_v" = Source[Schema="dbo",Item="PL_template_store_cc"][Data],
#"Filtered Rows" = Table.SelectRows(#"dbo_fpa-channel_store_v", each ([CC] <> null)),
#"Added Custom" = Table.AddColumn(#"Filtered Rows", "StoreCC", each Text.Combine([CC], [StoreName]), ": "),
#"Changed Type" = Table.TransformColumnTypes(#"Added Custom",{{"CC", Int64.Type}}),
#"Filtered Rows1" = Table.SelectRows(#"Changed Type", each ([StoreFormatName] <> "New Channel"))
in
#"Filtered Rows1"

```

✓ No syntax errors have been detected.

Done Cancel

ภาพที่ 3.74 ภาพแสดงการนำเข้าข้อมูลของตาราง DIM-Store

4) ตาราง TB เป็นตารางที่เก็บข้อมูลรายได้และค่าใช้จ่าย ซึ่งรายงานกำไรขาดทุนที่ได้จัดทำ 2 ประเภทมีการจัดการของมูลที่ต่างกันในระดับต้นนี้

การจัดการกับข้อมูลของรายงานงบกำไรขาดทุนแบบ KPI เป็นดังนี้

- นำเข้าข้อมูลมาจากรฐานข้อมูล FIN จากตาราง fpa\_channel\_pl\_snapshot\_v
- สร้างคอลัมน์เพิ่ม โดยคัดลอกคอลัมน์ CC มาตั้งชื่อว่า CC\_number และเปลี่ยนประเภทข้อมูลของคอลัมน์ให้เป็น Int
- จัดลำดับของคอลัมน์ใหม่
- กรองข้อมูลเอาเฉพาะข้อมูลที่มีคอลัมน์ Format เป็น Department Store, Express, Extra, Hypermarket และ Supermarket
- กรองข้อมูลไม่เอาข้อมูลที่มีคอลัมน์ Acc เป็น 35554, 40002, 40008, 51801, 51802, 51804, 51805, 51808, 52911, 78902, 78904, 78905, 40001, 40004, 40005, 52901, 52904, 52906, 54601, 78910, 73240, 73232, 68445 และ 68345
- กรองข้อมูลไม่เอาข้อมูลที่มีคอลัมน์ Local เป็น 00930 และ 00932
- กรองข้อมูลไม่เอาข้อมูลที่มีคอลัมน์ CC เป็น 35900, 00016, 00020, 00061, 00001, 00007, 00010, 00012, 00056, 00057, 00058, 00059, 00060, 00090, 00013, 00055, 00011, 00015, 00014 และ 00040

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

TB

Display Options ?

```

let
Source = Sql.Database("THYVDRASQLH001", "FIN", [CommandTimeout=#duration(0, 1, 30, 0)]),
#"dbo_fpa-channel-all_trial_balance_detail_kpi_v" = Source([Schema="dbo",Item="fpa_channel_pl_snapshot_v"])(Data),
#"Duplicated Column" = Table.DuplicateColumn(#"dbo_fpa-channel-all_trial_balance_detail_kpi_v", "CC", "CC - Copy"),
#"Renamed Columns" = Table.RenameColumns(#"Duplicated Column",{{"CC - Copy", "CC_number"}}),
#"Changed Type" = Table.TransformColumnTypes(#"Renamed Columns",{{"CC_number", Int64.Type}},),
#"Reordered Columns" = Table.ReorderColumns(#"Changed Type",{"Acc", "CC", "CCI", "PRD", "PRJ", "Local", "Inter", "Year", "Period", "Period_Activity", "Period_Activity_YTD", "Form"},),
#"Reordered Rows" = Table.SelectRows(#"Reordered Columns", each [Format] = "Department Store" or [Format] = "Express" or [Format] = "Extra" or [Format] = "Hypermarket" or [Format] = "Supermarket"),
#"Filtered Rows1" = Table.SelectRows(#"Reordered Rows", each ([Acc] <> "35554" and [Acc] <> "40002" and [Acc] <> "40008" and [Acc] <> "51801" and [Acc] <> "51802" and [Acc] <> "51803" and [Local] <> "00930" and [Local] <> "00932" and ([CC] <> "35900" and [CC] <> "00016" and [CC] <> "00020" and [CC] <> "00061" and [CC] <> "00001" and [CC] <> "00007"
in
#"Filtered Rows1"

```

✓ No syntax errors have been detected.

Done Cancel

### ภาพที่ 3.75 ภาพแสดงการนำเข้าข้อมูลของตาราง TB แบบ KPI

การจัดการกับข้อมูลของรายงานงบกำไรขาดทุนแบบ Full set เป็นดังนี้

- นำเข้าข้อมูลมาจากฐานข้อมูล FIN จากตาราง fpa\_channel\_pl\_allocate\_snapshot\_v
- สร้างคอลัมน์เพิ่ม โดยคัดลอกคอลัมน์ CC มาตั้งชื่อว่า CC\_number และเปลี่ยนประเภทข้อมูลของคอลัมน์ให้เป็น Int
- จัดลำดับของคอลัมน์ใหม่
- กรองข้อมูลเอาเฉพาะข้อมูลที่มีคอลัมน์ Format เป็น Express, Express-Pool, Extra, Extra-Pool, Department Store, Department Store-Pool, Hypermarket, Supermarket และ Supermarket-Pool
- กรองข้อมูลไม่เอาข้อมูลที่มีคอลัมน์ CC เป็น 35900

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

Advanced Editor

TB

```
let
Source = Sql.Database("THYVDRASQLH001", "FIN", [CommandTimeout=#duration(0, 1, 30, 0)]),
#"dbo_fpa-channel-all_trial_balance_detail_kpi_v" = Source([Schema="dbo",Item="fpa_channel_pl_allocate_snapshot_v"])[Data],
#"Duplicated Column" = Table.DuplicateColumn(#"dbo_fpa-channel-all_trial_balance_detail_kpi_v", "CC", "CC - Copy"),
#"Renamed Columns" = Table.RenameColumns(#"Duplicated Column",{{"CC - Copy", "CC_number"}}),
#"Changed Type" = Table.TransformColumnTypes(#"Renamed Columns",{{"CC_number", Int64.Type}}),
#"Reordered Columns" = Table.ReorderColumns(#"Changed Type",{"Acc", "CC", "CCT", "PRD", "PRJ", "Local", "Inter", "Year", "Period", "Period_Activity", "Period_Activity_YTD", "Form"},
#"Filtered Rows" = Table.SelectRows(#"Reordered Columns", each [Format] <> null and [Format] <> "Cost of store closure" and [Format] <> "Pre-opening" and [Format] <> "UNKNOWN" and
#"Filtered Rows1" = Table.SelectRows(#"Filtered Rows", each (([Format] = "Department Store" or [Format] = "Department Store-Pool" or [Format] = "Express" or [Format] = "Express-P"
and [CC]<="35900"))
in
#"Filtered Rows1"
```

✓ No syntax errors have been detected.

Done Cancel

ภาพที่ 3.76 ภาพแสดงการนำเข้าข้อมูลของตาราง TB แบบ Full set

- 5) ตาราง M-PL-Template-Store-1-Order เป็นตารางที่เก็บข้อมูลชื่อประเภทรายได้และค่าใช้จ่ายระดับที่ 1
- นำเข้าข้อมูลมาจากรานข้อมูล LOCALMASTER จากตาราง PL\_template\_store\_1\_order

Advanced Editor

M-PL-Template-Store-1-Order

```
let
Source = Sql.Database("THYVDRASQLH001", "LOCALMASTER"),
#"pl_template_store_1_order" = Source([Schema="dbo",Item="pl_template_store_1_order"])[Data]
in
#"pl_template_store_1_order"
```

✓ No syntax errors have been detected.

Done Cancel

ภาพที่ 3.77 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Template-Store-1-Order

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 6) ตาราง M-PL-Template-Store-1-1-Order เป็นตารางที่เก็บข้อมูลชื่อประเภทรายได้และค่าใช้จ่ายระดับที่ 2
- นำเข้าข้อมูลมาจากรฐานข้อมูล LOCALMASTER จากตาราง PL\_template\_store\_1.1\_order



ภาพที่ 3.78 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Template-Store-1-1-Order

- 7) ตาราง M-PL-Template-Store-1-1-1-Order เป็นตารางที่เก็บข้อมูลชื่อประเภทรายได้และค่าใช้จ่ายระดับที่ 3
- นำเข้าข้อมูลมาจากรฐานข้อมูล LOCALMASTER จากตาราง PL\_template\_store\_1.1.1\_order



ภาพที่ 3.79 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Template-Store-1-1-1-Order

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเว็บไซต์ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- 8) ตาราง M-PL-Store เป็นตารางที่เก็บข้อมูลลำดับชั้นของชื่อประเภทรายได้และค่าใช้จ่าย
- นำเข้าข้อมูลมาจากฐานข้อมูล LOCALMASTER จากตาราง PL\_template\_store

#### M-PL-Store

```
let
Source = Sql.Database("THTVORASQLH001", "LOCALMASTER"),
dbo_PL_template_store = Source[Schema=dbo,Item="PL_template_store"][Data]
in
dbo_PL_template_store
```

✓ No syntax errors have been detected.

Done Cancel

ภาพที่ 3.80 ภาพแสดงการนำเข้าข้อมูลของตาราง M-PL-Store

- 9) ตาราง M-Format-Hierarchy เป็นตารางที่เก็บข้อมูลลำดับชั้นของชื่อประเภทร้านค้าปลีก
- นำเข้าข้อมูลมาจากฐานข้อมูล LOCALMASTER จากตาราง PL\_format\_hierarchy
  - กรองข้อมูลไม่เอาข้อมูลที่มีคอลัมน์ 1.1.1 Format เป็น Market Place

#### M-Format-Hierarchy

```
let
Source = Sql.Database("THTVORASQLH001", "LOCALMASTER"),
#pl_format_hierarchy = Source[Schema=dbo,Item="PL_format_hierarchy"][Data],
#"Filtered Rows" = Table.SelectRows(pl_format_hierarchy, each ([1.1.1 Format] <> "Market Place"))
in
#"Filtered Rows"
```

✓ No syntax errors have been detected.

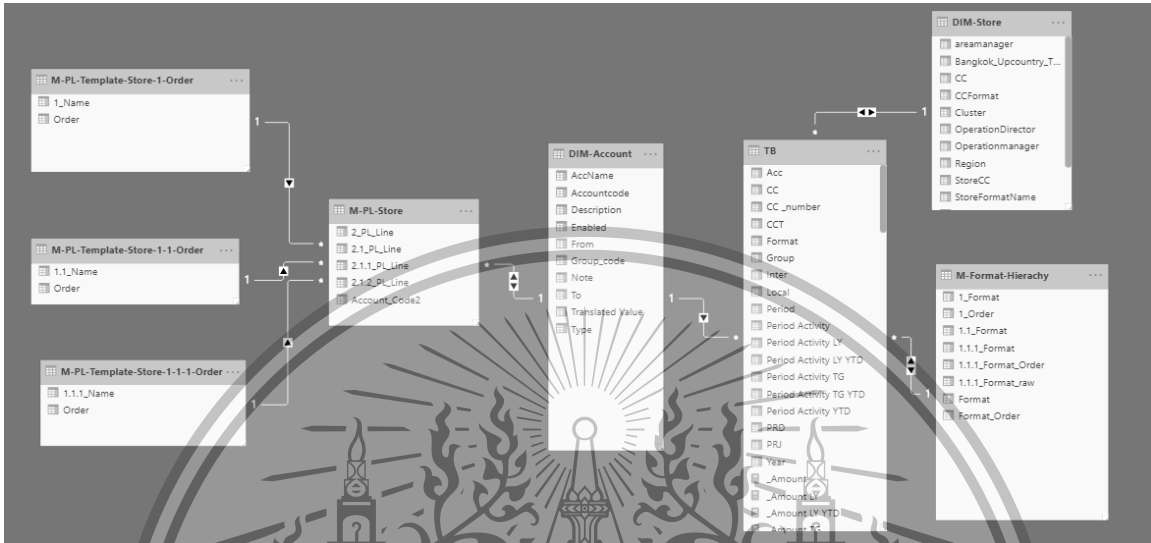
Done Cancel

ภาพที่ 3.81 ภาพแสดงการนำเข้าข้อมูลของตาราง M-Format-Hierarchy

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5.2 สร้างเส้นความสัมพันธ์ระหว่างตาราง

ใน Microsoft Power BI สร้างเส้นความสัมพันธ์ระหว่างแต่ละตาราง เพื่อให้สามารถแสดงข้อมูลได้อย่างถูกต้องจากทุกมุมมอง



ภาพที่ 3.82 ภาพแสดงความสัมพันธ์ของตารางใน Microsoft Power BI

### 3.5.3 จัดทำรายงานงบกำไรขาดทุน

เลือกรูปแบบในการแสดงผลในเหมาะสมกับข้อมูลที่มี โดยเลือกการแสดงผลเป็นแบบ Matrix ซึ่งจะแสดงข้อมูล Actual และ Target ในมุมมองของ Account group หรือประเภทของรายได้และค่าใช้จ่าย

Description	Current Period				Year to Date			
	Actual	%	Var TG	%Var TG	Actual	%	Var TG	%Var TG
Net Sale excl.VAT								
Net Margin								
Stock Loss								
Payroll								
Store Expense								
Property Cost								
Centrally cost								
Operating Profit								

ภาพที่ 3.83 ภาพแสดงตัวอย่างหน้ารายงานงบกำไรขาดทุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีการสร้างคอลัมน์ใหม่ขึ้นที่ Microsoft Power BI เป็นคอลัมน์ที่เกี่ยวข้องกับเปอร์เซ็นต์ สร้างคอลัมน์โดยเขียน DAX คำนวณหาเปอร์เซ็นต์ ใช้โค้ดในรูปแบบเดียวกันทุกคอลัมน์

```

1 %Actual =
2 var a = CALCULATE(TB[Actual],FILTER(ALL('M-PL-Store'),'M-PL-Store'[2_PL_Line] = "Net Sale excl.Vat"))
3 var b = CALCULATE(TB[Actual],FILTER(ALL('M-PL-Store'),'M-PL-Store'[2.1.1_PL_Line] = "Net Sales Ambient
  excl. VAT"))
4 var c = CALCULATE(TB[Actual],FILTER(ALL('M-PL-Store'),'M-PL-Store'[2.1.1_PL_Line] = "Net Sales Fresh
  Food excl. VAT"))
5 var ans1 = TB[Actual]/a
6 var ans2 = TB[Actual]/b
7 var ans3 = TB[Actual]/c
8 |
9 return
10 IF(SELECTEDVALUE('M-PL-Store'[2.1.1_PL_Line]) = "Ambient Damage and Out of Code" || SELECTEDVALUE
  ('M-PL-Store'[2.1.1_PL_Line]) = "Unknown Loss (Shrinkage)", ans2
11 , IF(SELECTEDVALUE('M-PL-Store'[2.1.1_PL_Line]) = "Fresh Food Damage and Out of Code" || SELECTEDVALUE
  ('M-PL-Store'[2.1.1_PL_Line]) = "Fresh Operations Loss (FOL)", ans3, ans1))
  
```

01007	15,454	22.70%	15,454	30%	1,248	3.03%	15,454	22.70%	15,254	NAN	-136,280	4.52%
01008	15,308	21.03%	15,308	NAN	694	2.99%	15,308	21.03%	15,308	NAN	-146,656	4.90%

ภาพที่ 3.84 ภาพแสดงโค้ดที่ใช้หาค่าเปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

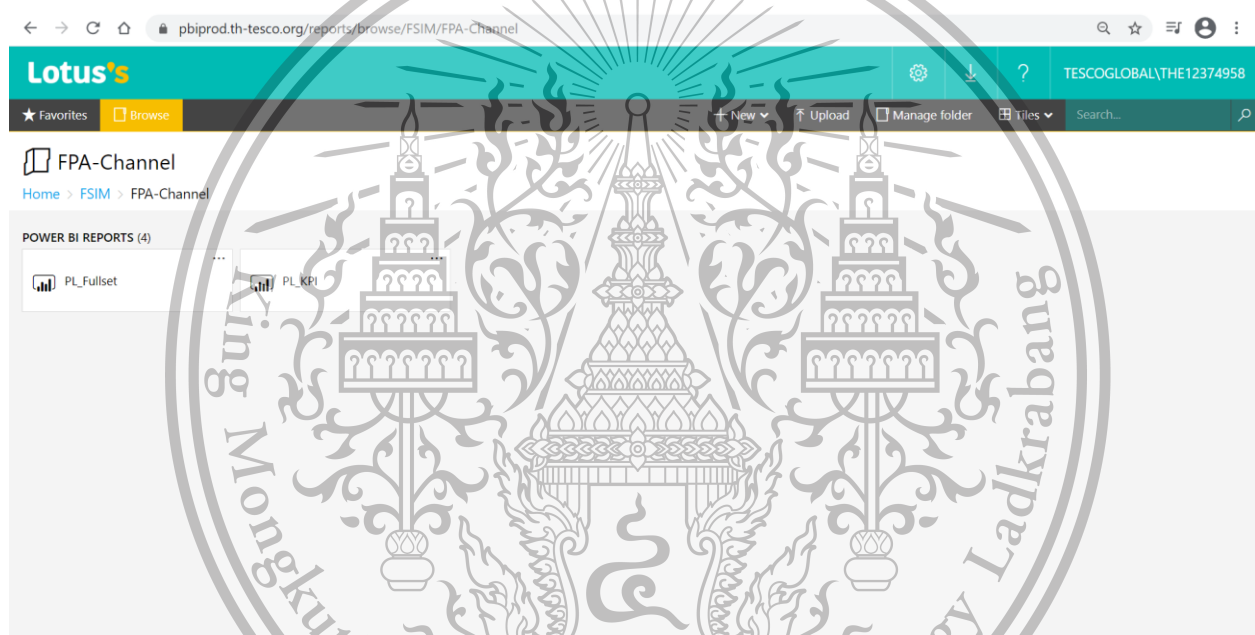
Forbidden to modify the content, and cite the document when use.

## บทที่ 4

### ผลการดำเนินโครงการ

#### 4.1 วิธีการเข้าถึงรายงานงบกำไรขาดทุน

รายงานงบกำไรขาดทุนจะถูกอัปโหลดเก็บไว้ที่ Microsoft Power BI Server ของทางบริษัท ซึ่งโปรแกรมนี้สามารถรองรับการใช้งานได้หลากหลายทั้งคอมพิวเตอร์ แล็ปท็อป แท็บเล็ต และสมาร์ทโฟน ทำให้ผู้ใช้งานไม่ว่าจะเป็นผู้ใช้งานที่ประจำอยู่ที่สำนักงาน หรือผู้ใช้งานที่ประจำอยู่ที่ร้านค้าปลีกนั้นสามารถเปิดใช้งานรายงานงบกำไรขาดทุนได้อย่างสะดวก รวดเร็ว โดยเข้าถึงรายงานได้ผ่านอุปกรณ์ที่รองรับ Microsoft Power BI Server เหล่านี้ได้ทั้งหมด ซึ่งมีการแบ่งเป็น 2 ไฟล์ ตามประเภทของรายงาน



ภาพที่ 4.1 ภาพแสดงหน้าต่างของ Microsoft Power BI Server

การใช้งานผ่าน Microsoft Power BI Server นอกจากจะออกแบบมาให้ผู้ใช้งานสามารถใช้เข้าถึงรายงานงบกำไรขาดทุนได้อย่างสะดวกแล้ว ผู้ใช้งานยังได้รับข้อมูลที่มีการอัปเดตให้เป็นข้อมูลล่าสุดอยู่เสมอ เนื่องจากการตั้งค่าให้รายงานรีเฟรชข้อมูลเป็นประจำทุกวัน ทำให้ช่วยเพิ่มประสิทธิภาพในการนำเอารายงานงบกำไรขาดทุนนี้ไปบริหารกิจการต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.



## 4.2 รายงานงบกำไรขาดทุน

### 4.2.1 หน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับแสดงผลบน server

ในหน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับแสดงผลบน server นั้นจะสามารถแตกลงไปดูข้อมูลได้ทั้งหมด 5 ระดับ ตามภาพที่ 4.2 และมีการคำนวณค่า Retail Contribution exclude Mall and Other income, Retail Contribution include Mall and Other income และ OPEX

Rows

Description	✓	✕
1.1_Name	✓	✕
1.1.1_Name	✓	✕
2.1.2_PL_Line	✓	✕
AccName	✓	✕

ภาพที่ 4.2 ภาพแสดงระดับของข้อมูลสำหรับแสดงผลบน server

Description	Current Period						Year to Date					
	Actual	%	Var TG	%Var TG	Var LY	%Var LY	Actual	%	Var TG	%Var TG	Var LY	%Var LY
Net Sale excl.VAT												
Net Margin												
Stock Loss												
Payroll												
Store Expense												
Property Cost												
Mail & Retail Service												
Centrally cost												
Operating Profit												
Retail Contribution excl. Mall & Other income												
Retail Contribution inc. Mall & Other income												
OPEX												

ภาพที่ 4.3 ภาพแสดงหน้ารายงานงบกำไรขาดทุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.2 หน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับการ export ข้อมูลออกมาเป็นไฟล์ csv

ในหน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับการ export ข้อมูลออกมาเป็นไฟล์ csv นั้นจะสามารถตกลงไปดูข้อมูลได้ทั้งหมด 8 ระดับ ตามภาพที่ 4.4

Rows	
Description	✓ X
1.1_Name	✓ X
1.1.1_Name	✓ X
2.1_PL_Line	✓ X
CC	✓ X
StoreCC	✓ X
areamanager	✓ X
Operationmanager	✓ X

ภาพที่ 4.4 ภาพแสดงระดับของข้อมูลสำหรับ export

Description	Current Period					Year to Date						
	Actual	%	Var TG	%Var TG	Var LY	%Var LY	Actual	%	Var TG	%Var TG	Var LY	%Var LY
Net Sale excl.VAT												
Net Sales excl.VAT												
Net Sales Fresh Food excl.VAT												
00020												
00040												
01000												
01001												
01002												
01003												
01004												
01005												
01006												
01007												
01008												
01009												
01010												
01011												
01012												
01013												
01014												
01015												
01016												
01017												
01018												
01019												
01020												

ภาพที่ 4.5 ภาพแสดงหน้ารายงานงบกำไรขาดทุนสำหรับการ export

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.3 หน้ารายงานงบกำไรขาดทุนที่สร้างสำหรับปรีนลงกระดาษ A3

Store Profit & Loss Statement

2021 03, ...

Region: All | OD: All | OM: All | AM: All | Format: All | Store: All

Description	Current Period						Year to Date					
	Actual	%	Ver TG	S/Wr TG	Ver LF	S/Wr LF	Actual	%	Ver TG	S/Wr TG	Ver LF	S/Wr LF
Net Sale excl.VAT												
Net Sale incl.VAT												
Net Margin												
Net Margin												
Stock Loss												
Payroll												
Store Expense												
Retail FM Expense												
Bank Charge												
Property Cost												
Mail & Retail Service												
Net Marketing												
Net DC												
Operating Profit												
Retail Contribution excl. Mail & Other Income												
Retail Contribution Inc. Mail & Other Income												
OPEX												

ภาพที่ 4.6 ภาพแสดงหน้ารายงานงบกำไรขาดทุนสำหรับปรีน A3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

## บทที่ 5

### สรุปผลการดำเนินโครงการ

#### 5.1 สรุปผล

จากการพัฒนาระบบสร้างรายงานงบกำไรขาดทุนแบบอัตโนมัติเสร็จสิ้นนั้น ได้ทดสอบกระบวนการสร้างรายงาน และนำไปเปรียบเทียบกับกระบวนการสร้างรายงานแบบเก่าซึ่งเป็นการสร้างรายงานแบบใช้แรงงานคนในการจัดทำนั้น พบว่าสามารถแก้ปัญหาที่ผู้ใช้งานเคยประสบได้ทั้งหมด

- 1) ปัญหาเรื่องเวลาที่ใช้ในการจัดทำรายงานงบกำไรขาดทุน
- 2) ปัญหาเรื่องความถูกต้องแม่นยำของข้อมูลที่นำมาแสดงผล
- 3) ปัญหาเรื่องข้อจำกัดของโปรแกรม Microsoft Excel ที่ไม่สามารถรองรับข้อมูลทั้งหมดได้ จึงต้องแบ่งรายงานงบกำไรขาดทุนออกเป็น 4 ไฟล์

โดยสามารถเปรียบเทียบประสิทธิภาพของกระบวนการสร้างรายงานงบกำไรขาดทุนแบบเก่าและแบบที่พัฒนาขึ้นมาใหม่ ดังตารางที่ 5.1

ตารางที่ 5.1 ตารางเปรียบเทียบประสิทธิภาพระหว่างกระบวนการสร้างรายงานแบบเก่าและแบบใหม่

	แบบเก่า	แบบใหม่	การเปลี่ยนแปลง
เวลาที่ใช้ในการจัดทำงบกำไรขาดทุน	48 ชั่วโมง / เดือน	2 ชั่วโมง / เดือน	เวลาลดลง 46 ชั่วโมง (ลดลง 96%)
ความถูกต้องของข้อมูล	85% – 95%	95%-100%	ข้อมูลถูกต้องมากขึ้น (ความถูกต้องเพิ่มประมาณ 7.5%)
จำนวนไฟล์ในการแสดงผล	4 ไฟล์	2 ไฟล์	ไฟล์ลดลง 2 ไฟล์ (ลดลง 50%)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เวลาที่ใช้ในการจัดทำรายงานงบกำไรขาดทุน

ปัญหาเรื่องเวลาที่ใช้ในการจัดทำรายงานงบกำไรขาดทุนนั้นเป็นปัญหาที่สำคัญที่สุดที่จะต้องได้รับการแก้ไข กระบวนการจัดทำรายงานงบกำไรขาดทุนเดิมผู้ใช้งานเป็นผู้จัดทำเอง โดยการดึงข้อมูลจำนวนมาก และนำไฟล์ข้อมูลเหล่านั้นมาสร้างรายงานในโปรแกรม Microsoft Excel กระบวนการจัดทำรายงานจำเป็นต้องใช้บุคลากรจัดทำทั้งหมด 3 คน ใช้เวลาคนละประมาณ 16 ชั่วโมง ถือว่าใช้เวลามากเกินไปจนเกิดความจำเป็น ส่งผลต่อประสิทธิภาพในการทำงานในระยะยาว ซึ่งหลังจากปรับปรุงและพัฒนาระบบให้สามารถสร้างรายงานงบกำไรขาดทุนแบบอัตโนมัติ ทำให้ผู้ใช้งานไม่จำเป็นต้องจัดทำรายงาน และสามารถลดเวลาที่ใช้ในการจัดทำรายงานเหลือเพียงแค่ 2 ชั่วโมงเท่านั้น

- ความถูกต้องของข้อมูล

ปัญหาที่สำคัญรองมาจากเรื่องของเวลาที่ใช้ในการจัดทำคือปัญหาความถูกต้องของข้อมูล เนื่องจากแต่เดิมนั้นรายงานงบกำไรขาดทุนได้ถูกจัดทำขึ้นจากผู้ใช้งานด้วยกระบวนการแมนนวลทั้งหมด ดังนั้นทำให้เกิดข้อผิดพลาดที่เกิดจากมนุษย์ขึ้นเป็นประจำ ซึ่งหลังจากปรับปรุงและพัฒนาระบบให้สามารถสร้างรายงานงบกำไรขาดทุนแบบอัตโนมัติ มีการตัดกระบวนการที่จัดทำด้วยมนุษย์ออกทั้งหมด ทำให้ได้ข้อมูลที่ถูกต้อง แม่นยำ ไม่เกิดข้อผิดพลาดที่เกิดจากมนุษย์

- โปรแกรมที่ใช้ในการแสดงรายงานงบกำไรขาดทุน

รายงานงบกำไรขาดทุนแบบเดิมนั้นจะถูกจัดทำขึ้นในโปรแกรม Microsoft Excel ซึ่งไม่สามารถรองรับข้อมูลของรายงานงบกำไรขาดทุนได้ทั้งหมด ทำให้ต้องแสดงผลในหลายไฟล์ ซึ่งหลังจากเปลี่ยนให้รายงานงบกำไรขาดทุนแบบอัตโนมัติมาแสดงผลที่โปรแกรม Microsoft Power BI ทำให้สามารถรองรับข้อมูลได้มากยิ่งขึ้น และหน้าแสดงผลอยู่ในรูปแบบที่อ่านง่าย สวยงาม ดูสบายตา ซึ่งเป็นผลดีต่อการนำรายงานงบกำไรขาดทุนนี้ไปวิเคราะห์ และปรับปรุงการดำเนินงานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## เอกสารอ้างอิง

- [1] Ek-chai Distribution System Co. Ltd [TH]. เกี่ยวกับเรา ประวัติความเป็นมา [Online].  
แหล่งข้อมูล: <https://www.tescolotus.com/about/history>  
ค้นหาข้อมูลวันที่ 26 ธันวาคม 2563
- [2] thaiwinner. Income Statement [Online].  
แหล่งข้อมูล: <https://thaiwinner.com/income-statement/>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [3] Mindphp. Database คืออะไร ระบบฐานข้อมูล คือ การเก็บรวบรวมข้อมูลไว้ด้วยกัน อย่างมีระบบ [Online].  
แหล่งข้อมูล: <https://www.mindphp.com/68-php-e-commerce/2055-database>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [4] Sanook. เครือข่ายคอมพิวเตอร์ [Online].  
แหล่งข้อมูล: <https://guru.sanook.com/2287/>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [5] Mango. Server คืออะไร ทำหน้าที่อะไร มีประโยชน์อย่างไร Server มีกี่ประเภท [Online].  
แหล่งข้อมูล: <https://www.mangoconsultant.com/th/news-knowledge/knowledge/>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [6] Mindphp. Batch File (แบทไฟล์) คืออะไร [Online].  
แหล่งข้อมูล: <https://mindphp.com/forums/viewtopic.php?f=79&t=39086>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [7] Mindphp. Window คืออะไร วินโดวส์ คือระบบปฏิบัติการระบบหนึ่ง [Online].  
แหล่งข้อมูล: <https://www.mindphp.com/คู่มือ/73-คืออะไร/2124-windows-คืออะไร.html>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [8] Mindphp. SQL Server คืออะไร [Online].  
แหล่งข้อมูล: [https://www.bestinternet.co.th/single\\_blog.php?id=43&Microsoft%20SQL](https://www.bestinternet.co.th/single_blog.php?id=43&Microsoft%20SQL)  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564
- [9] Thepexcel. วิธี Export ข้อมูลจาก Power Query [Online].  
แหล่งข้อมูล: <https://www.thepexcel.com/export-power-query-million-rows-text-file/>  
ค้นหาข้อมูลวันที่ 11 มกราคม 2564

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This material is reserved for educational use only, not allowed for commercial use.

Forbidden to modify the content, and cite the document when use.

# Profit and Loss Reports Automation

Haruethai Amnatchongcharoon

Advisor name : Asst.Prof.Dr. Vanvisa Chutchavong

## Abstract

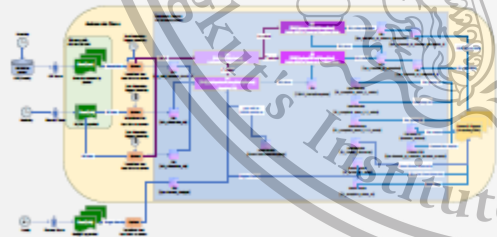
The profit and loss reports are important part of verifying the status of a company. See earnings as well as administrative planning, helping businesses to keep going. But nowadays, the process of generating profit and loss reports in C.P. Retail Holding CO., LTD. is facing a problem with time in the reporting process. The information used comes from many sources and additional calculations are required, causing frequent errors. In addition, the data used as large data cannot be reported in a single Microsoft Excel file. These problems have resulted by improving the reporting to be automatic and generate reports in Microsoft Power BI.

## Introduction

The process of generating profit and loss reports in C.P. Retail Holding CO., LTD. is using too much human resource and time resource and some part of data is incorrect by human errors and cannot create the reports in 2 files (P&L NPI and P&L Full set) because of excel limitation. This project makes a automate process for create profit and loss reports, reducing time and human resource for creating profit and loss reports, reducing human errors and dashboard created by Microsoft Power BI makes user easy to use.

## Methodology

To create the profit and loss reports, the process start with getting data in system and data provided from financial team. These data will get through multiple processes in multiple tools and need to create database to stored the data. Finally, these data will display in the Microsoft Power BI dashboard.



- Step 1 : Get raw data in the share path, provided by OFI and financial teams. Most of the data will be provided once a month.
- Step 2 : Using Python scripts to automatically transform data and load data into database server.
- Step 3 : In SSMS, transform data again and mapping data with others table in same database and difference database by using stored procedure.
- Step 4 : Prepare data in SSMS by create views.
- Step 5 : Connect Microsoft Power BI with data that was prepare in the view and other necessary table in the database.
- Step 6 : Create new columns in Microsoft Power BI dashboard, using DAX to create column.



## Results

The profit and loss reports uploaded to Microsoft Power BI Server. Employee in both retail stores and head office can access the profit and loss reports by using computer, tablet, laptop or smartphone.



Profit and loss report for Lopburi display in matrix, data represented in group of income and expense types, having filter related to area of store.



## Conclusion

This project can meet user requirement, improve the reports creating process and output.

	Manual Version	Automate Version	Changed
Time used to create the reports	48 hours / month	2 hours / month	Decreased 46 hours
Data accurate	83% - 95%	93% - 100%	Increase accuracy
Number of files for display reports	4 files	2 files	Decreased 2 files

## References

- thaiwinner. Income Statement [Online]. Website : <https://thaiwinner.com/income-statement/> [11 January 2021]  
9expert. Power BI [Online]. Website : <https://www.9experttraining.com/articles/> [11 January 2021]

## Acknowledgement

This project co-operates with C.P. Retail Holding CO., LTD.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้